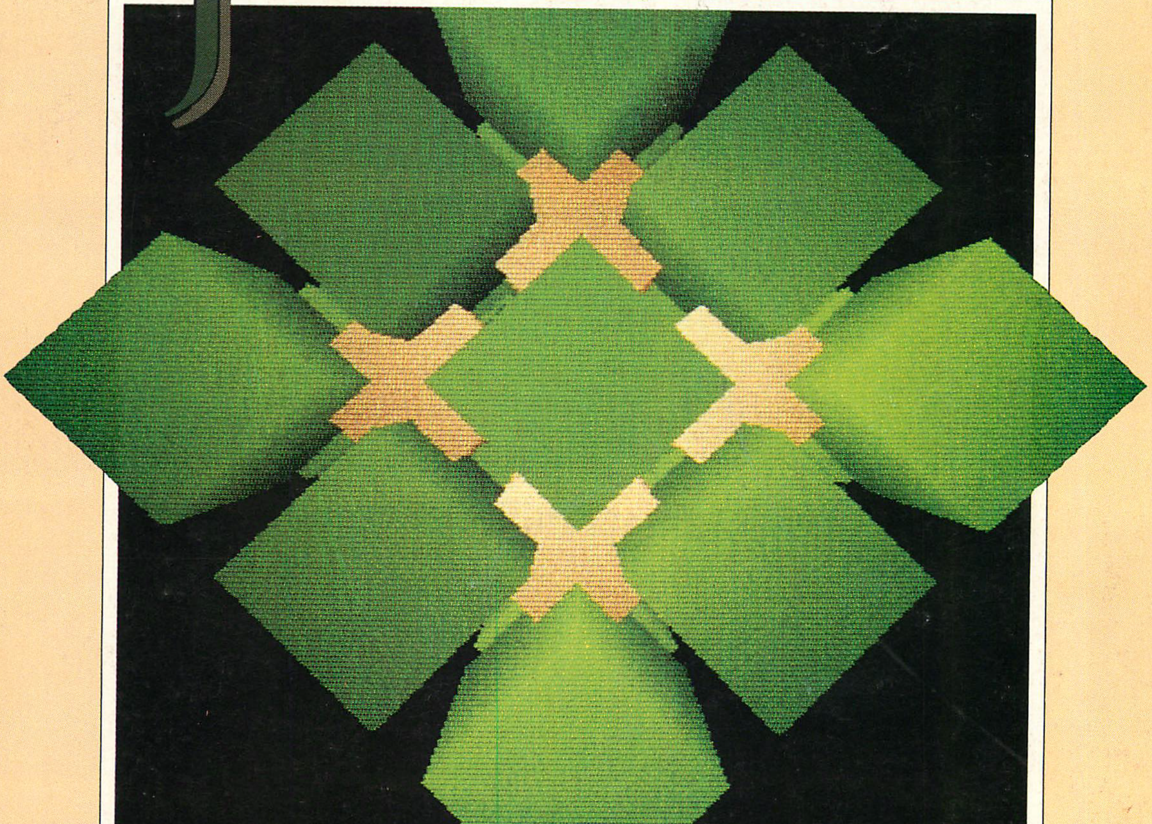
AUGUST 1985

VOL. 3, NO. 8 $3.95

FOR IBM PERSONAL COMPUTER USERS

PC TECH JOURNAL



DATA MANAGERS AS DEVELOPMENT TOOLS

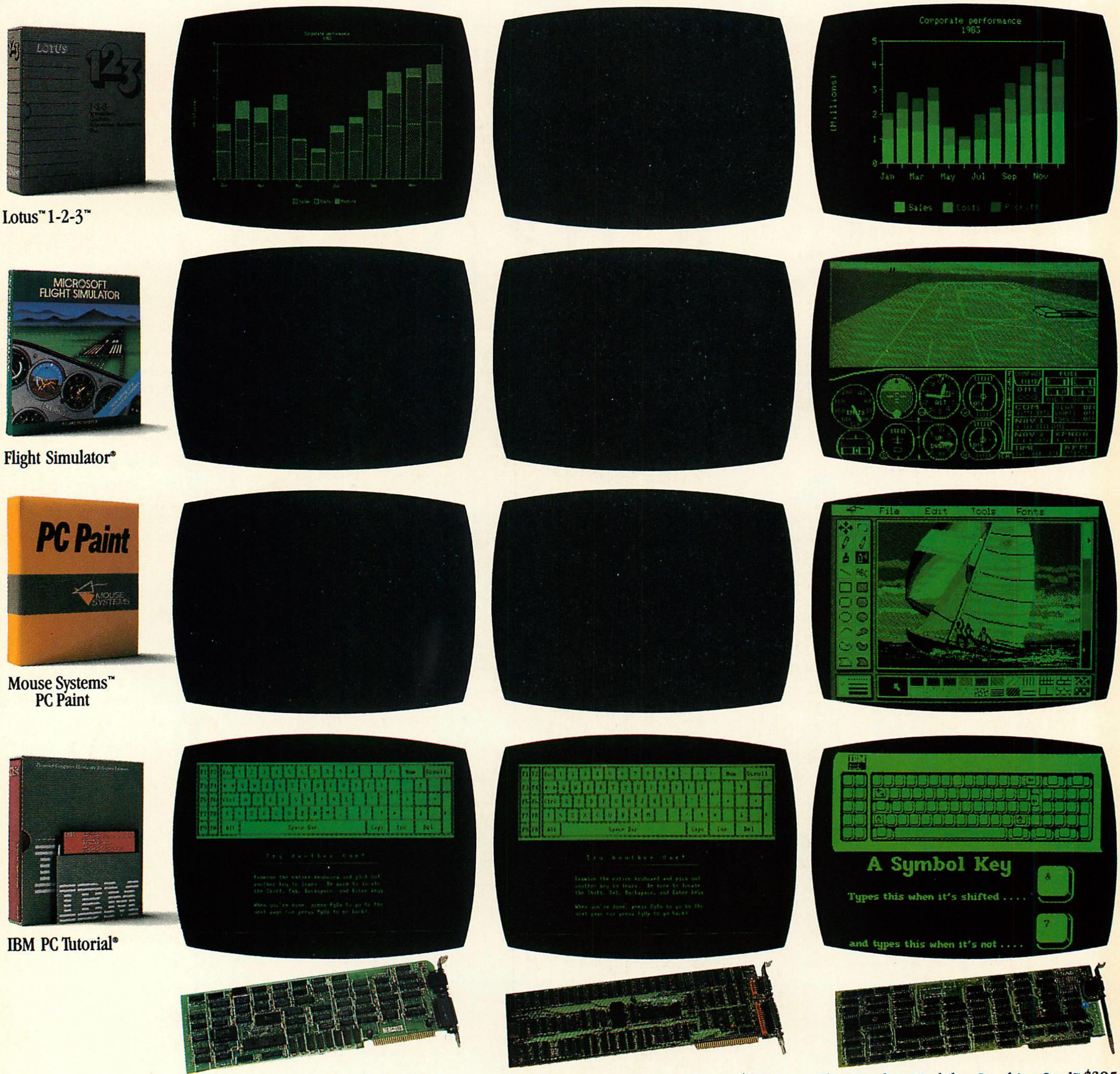
Evaluation Criteria

First in the series: DataFlex

[redacted]

IBM MONOCHROME GRAPHICS. NO COMPROMISE.

$50 Trade-In Allowance on your old Video Card*



Lotus™ 1-2-3™

Flight Simulator®

Mouse Systems™ PC Paint

IBM PC Tutorial®

The Hercules Graphics Card™ $499

The IBM® Monochrome Adapter $250

The Paradise Modular Graphics Card™ $395

The Paradise Modular Graphics Card is the *only* video card that runs all popular IBM color/graphics software—full-screen—on an IBM (or equivalent) monochrome monitor. *Unmodified.*

Right off the shelf.

And not just ordinary monochrome: Crisp, full-screen, flicker-free monochrome graphics translated in up to 16 shades.

- Includes IBM-compatible color graphics on an RGB or composite video monitor.
- Optional memory (up to 384Kb), clock calendar or floppy disk controller, and parallel, serial or game ports available.

All in a single PC or XT slot.

This may be the only card you'll ever need.

Call today for more information:
1-800-527-7977, In CA: 1-800-822-2020 (Dept. 999)
150 North Hill Drive, Brisbane, CA 94005, 415-468-6000

PARADISE

You can measure it.

Trademarks: Paradise Modular Graphics Card–Paradise Systems, Inc.; Hercules Graphics Card–Hercules Computer Technology; 1-2-3, Lotus–Lotus Development Corporation; Mouse Systems, PC Paint–Mouse Systems Corporation. Registered trademarks: IBM, PC Tutorial–International Business Machines Corporation; Flight Simulator–Microsoft Corporation.

*Paradise will send you a check for $50 when you send us your old Hercules Graphics Card or IBM Monochrome Card and the Completed Paradise Modular Graphics Card Warranty. Offer Expires 8/31/85.

CIRCLE NO. 230 ON READER SERVICE CARD

"Nature Imposes Few Restrictions on Those Daring Enough to Lead"

ZIM is a fully integrated fourth generation application development system designed for leading system integrators and corporate and independent applications developers.

*COMPLETE DEVELOPMENT ENVIRONMENT
- -Report Writer
- -Forms Painter and Manager
- -Data Dictionary
- -Application Generator
- -Non-procedural Programming Language
- -Compiler
- -C Language Interface
- -Runtime System

*POST RELATIONAL
- -Entity Relationship Model
- -Powerful extension of Relational Model

*MAINFRAME POWER, FUNCTIONALITY AND PERFORMANCE

*APPLICATIONS PORTABILITY
- -MS-DOS, UNIX, XENIX, and QNX

*MULTI-USER
- -Full transaction processing control

*NETWORKING

*APPLICATIONS LIMITED ONLY BY HARDWARE

*BUILT-IN STRATEGY OPTIMIZER

*ENGLISH-LIKE LANGUAGE

*QUALITY PRODUCT SUPPORT

ZIM is a mainframe system that runs on micro-computers and on super micro-computers. If you want mainframe power, speed, flexibility and freedom from arbitrary limitations all at a micro price, talk to us about an evaluation system.

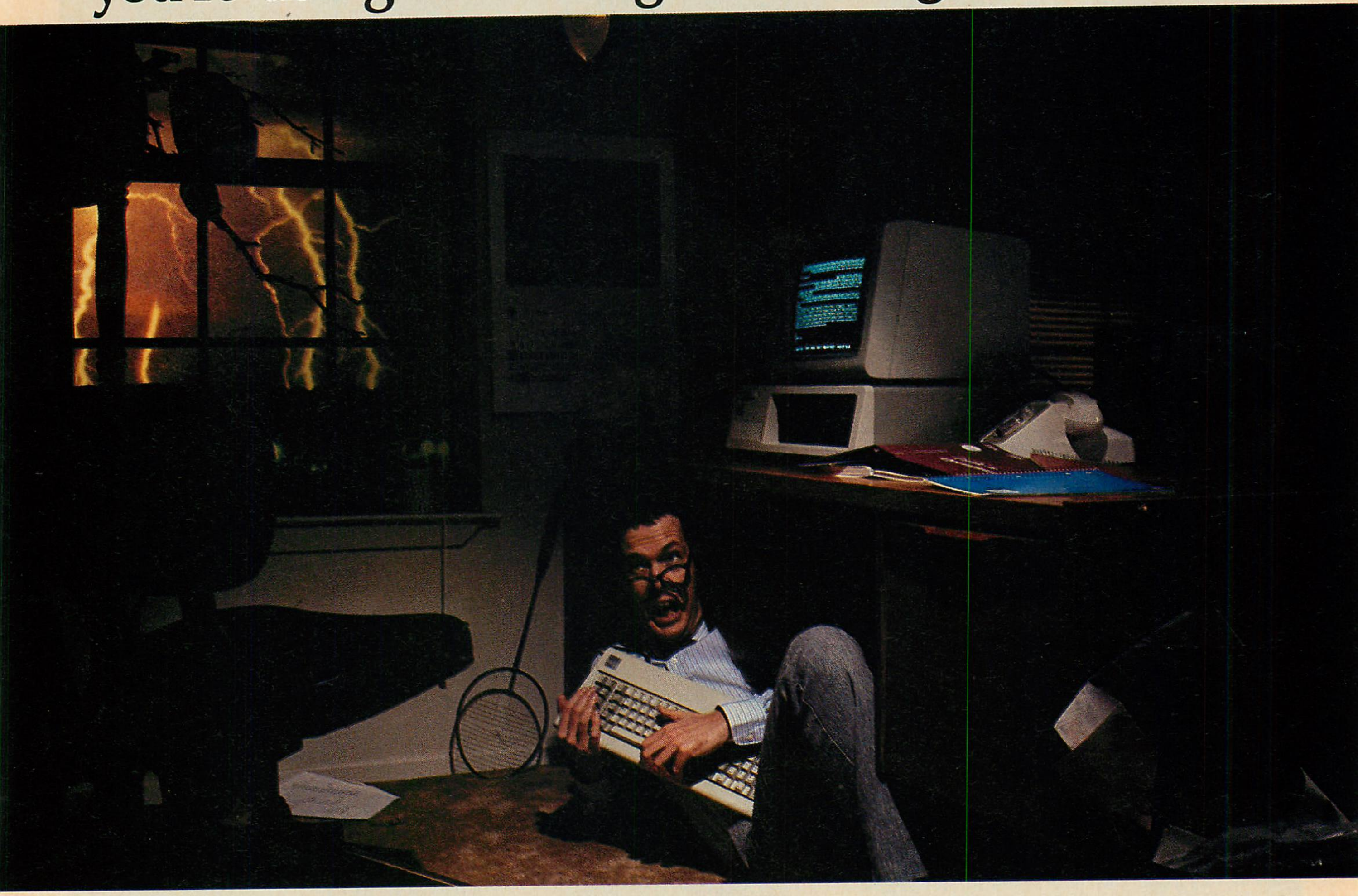
Dealer inquiries are welcome.

ZIM
The Information Interface

ZANTHE

1785 Woodward Dr., Ottawa, Ontario
K2C 0R1 (613)727-1397

If lightning still scares you, you're using the wrong file manager.



Be sure. Btrieve™.

Lightning may strike. But it doesn't have to destroy your database.

Btrieve™ file management offers automatic file recovery after a system crash. So accidents and power failures don't turn into database disasters. Your Btrieve-based applications will come up when the lights come back on.

Fast. Btrieve is lightning fast, too. Its b-tree file structure automatically balances—you never waste time reorganizing the index. And Btrieve is written in Assembly language especially for the IBM PC™. The result: electrifying speed on file maintenance routines. Applications that run fast. Users who don't waste time waiting.

The standard for networking. When your application requires multi-user file sharing, Btrieve/N (network version) sets the standard for the industry's most popular LANs: IBM's PC Network, Netware™, Davong MultiLink™, Omninet™, PC Net™, EtherSeries™, Nestar™, and NetOne™. Btrieve/N offers **safe** network file management that coordinates simultaneous updates and protects against lost data.

Fully-relational data management. Using SoftCraft's entire family of products gives you a complete, fully-relational database management system. Rtrieve™ adds report writing capabilities for generating the reports you need. Xtrieve™ speeds users through database queries with interactive, on-screen menus—no command language or special syntax.

For professional programmers. Btrieve is the fast, reliable answer for all your application development. In any development language—BASIC, Pascal, Cobol, C, Fortran, and APL. With multikey access to records. Unlimited records per file. Duplicate, modifiable, and segmented keys. Variable cache buffer.

With Btrieve, you can develop better applications faster. And know they'll be safe if lightning strikes.

Suggested retail prices: Btrieve, $245; Btrieve/N, $595; Xtrieve, $195; Xtrieve/N, $395; Rtrieve, $85; Rtrieve/N, $175. Requires PC-DOS or MS™-DOS 1.X, 2.X, or 3.X.

Btrieve, Xtrieve, and Rtrieve; IBM; Netware; Davong MultiLink; Omninet; PC Net; EtherSeries; Nestar; NetOne; and MS are trademarks of SoftCraft Inc.; International Business Machines; Novell Data Systems; Davong Systems Inc.; Corvus Systems; Orchid Technology; 3Com Corp.; Nestar Systems Inc.; Ungermann-Bass; and Microsoft Inc.

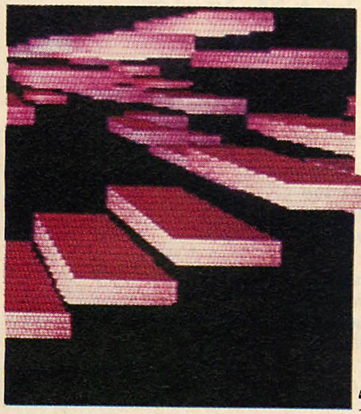
SoftCraft Inc.
P.O. Box 9802 #917
Austin, Texas 78766
(512) 346-8380 Telex 358 200

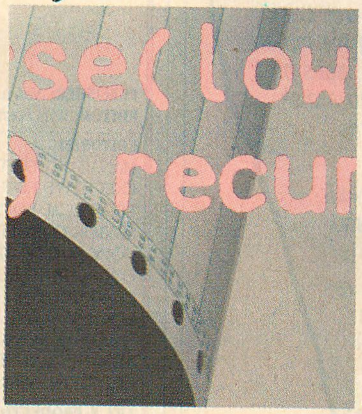
CIRCLE NO. 201 ON READER SERVICE CARD

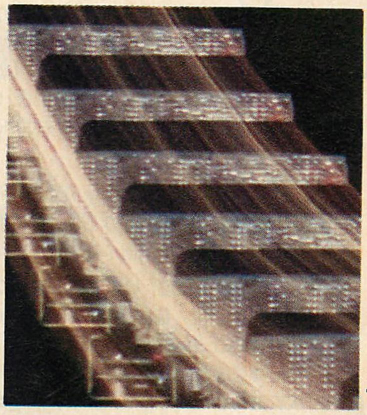
AUGUST 1985

PC TECH JOURNAL

VOLUME 3, NUMBER 8


52


82


74

Cover computer graphic · Dov Jacobson

PC TECH JOURNAL

VOL. 3, NO. 8

PUBLISHER: *Newton Barrett*
EDITOR: *Will Fastie*

EDITORIAL
MANAGING EDITOR: *Marjory Spraycar*
EXECUTIVE EDITOR: *Julie Anderson*
TECHNICAL EDITOR: *Jeff Duntemann*
SENIOR COPY EDITOR: *Susan Holly*
COPY EDITOR: *Gail Shaffer*
CONSULTING EDITOR: *Thomas V. Hoffmann*
CONTRIBUTING EDITORS: *Steven Armbrust, Don Awalt, Richard Foard, Ted Forgeron, Arthur A. Gleckler, Augie Hansen, Ted Mirecki, William H. Murray, Max Stul Oppenheimer*
ADMINISTRATIVE ASSISTANT: *Paula Lamberti*
EDITORIAL ASSISTANT: *Carole Autenzio*

ART & PRODUCTION
ART DIRECTOR: *Ina Saltz*
ASSOCIATE ART DIRECTOR: *Sharon Reuter*
ART ASSISTANT: *Sandra Ray*

ADVERTISING SALES
MARKETING COORDINATOR: *Julie Henderson*
ADVERTISING COORDINATOR: *Michele Fischetti*
DISTRICT MANAGERS:
Rita Burke, Jan Schultz—East Coast
Lisa Kampfmann—Midwest
Ted Bahr, Bill Bush, Phyllis Egan—West Coast
ACCOUNT REPRESENTATIVES:
Pat Toohey—East Coast
Rosemarie Caruso—Midwest and East Coast
Pam Sigal, Arlene Steadman—West Coast

CIRCULATION
CIRCULATION MANAGER: *Shane Boel*
CIRCULATION SALES DEVELOPMENT: *Daniel Rosensweig*
MEDIA MANAGER: *Melinda Kendall*

COMPUTER PUBLICATIONS DIVISION
PRESIDENT: *Kenneth H. Koppel*
VICE PRESIDENT, Editorial: *Jonathan D. Lazarus*
VICE PRESIDENT, Production: *Baird Davis*
VICE PRESIDENT, Licensing & Special Projects: *Jerry Schneider*
VICE PRESIDENT, Creative Services: *Herbert Stern*
VICE PRESIDENT, Circulation: *Alicia Marie Ivans*
VICE PRESIDENT, Circulation Services: *James Ramaley*
VICE PRESIDENT: *Hugh Tietjen*
MARKETING MANAGER: *Ronnie Sonnenberg*
VICE PRESIDENT, Marketing Services: *Ann Pollak Adelman*
BUSINESS MANAGER: *Gary A. Gustafson*
EDITORIAL DIRECTOR: *Ernest F. Baxter*

ZIFF-DAVIS PUBLISHING
PRESIDENT, *Richard P. Friese;* SENIOR VICE PRESIDENT, *Philip T. Heffernan;* SENIOR VICE PRESIDENT, *Philip Sine;* VICE PRESIDENT, *William L. Phillips;* TREASURER, *Selwyn Taubman;* SECRETARY, *Bertram A. Abrams*

EDITORIAL OFFICE
PC TECH JOURNAL, The World Trade Center, Suite 211, Baltimore, MD 21202. 301/576-0770. MCIMail: PCTECH. PCTECHline: 301/576-PCTJ.

ADVERTISING OFFICES
(East Coast and Midwest) One Park Avenue, New York, NY 10016. 212/503-5185.
(West Coast) 3460 Wilshire Blvd., Los Angeles, CA 90010. 213/387-2100; 11 Davis Drive, Belmont, CA 94002. 415/598-2290.

SUBSCRIPTION INQUIRIES
PC TECH JOURNAL, P.O. Box 2968, Boulder, CO 80321. Subscription service: 303/447-9330. Back issues: send $7.00/copy to *PC TECH JOURNAL*, Box CN, 1914, Morristown, NJ 07960.

PC TECH JOURNAL (ISSN 0738-0194) is published monthly, $29.97 for one year, $52.97 for two years, $69.97 for three years. Additional postage $12 for Canada & Foreign by Ziff-Davis Publishing Company, One Park Avenue, New York, NY 10016. Second-Class Postage paid at New York, NY and at additional mailing offices. POSTMASTER: Send address changes or subscription inquiries to P.O. Box 2968, Boulder, CO 80321.

Another in a series of productivity notes on MS-DOS™ software from UniPress.

TEXT EDITING

UNIPRESS EMACS™

Subject: Multi-window full screen editor.

Multiple windows allow several files (or portions of the same file) to be edited simultaneously. Programmable through macros and the built-in compiled MLISP™ extension language.

Features:

- Famed Gosling Version.
- Extensible through the built-in MLISP programming language and macros.
- Dozens of source code MLISP functions; including C, Pascal and MLISP syntax checking.
- EMACS runs on TI-PC™ IBM-PC AT,™ DEC Rainbow™ or any other MS-DOS machine. Requires at least 384k.
- Run Lattice® C or PsMake™ in the background and EMACS will point to any errors for ease of debugging. PsMake is a UNIX™-style "make" utility to automate the process of building complex programs.
- Optional Carousel Tools: UNIX-like facilities including cat, cp, cd, logout, ls, mv, pwd, rm, set, sh and more.

Price:

EMACS	$475
One month trial	75
Available for UNIX and VMS.	
PsMake	179
Carousel Tools	149
Full System	1,299
Includes EMACS (object), PsMake, Lattice C, PHACT™ ISAM and Carousel Tools.	

COMPILER FOR THE 8086™ FAMILY

LATTICE® C COMPILER

Subject: Compiler for MS-DOS.

Lattice C Compiler is regarded as the finest compiler for MS-DOS and is running on thousands of 8086 systems.

Features:

- Runs on the IBM-PC™ under MS-DOS 1.0, 2.0 or 3.0
- Produces highly optimized code.
- Small, medium, compact and large address models available.
- Standard C library.
- PLINK – optional full function linkage editor including overlay and support.

Price:

Lattice C Compiler	$425
PLINK	425

ISAM FILE SYSTEM & REPORT WRITER

PHACT™

Subject: Powerful Keyed File Access for MS-DOS.

PHACT ISAM is a keyed B+ tree file manager providing easy access to and manipulation of records in a database.

Features:

- Supports fixed and variable length records (1-9999 bytes).
- Up to 9 alternate indices are supported.
- Record locking allows each record in the database to allow multiple simultaneous updates.
- Records can be accessed on full or partial key.
- Includes full Lattice linkable library and high-level functions.
- Optional, PHACT-rg, a powerful and flexible report generator which provides a high level, easy-to-use command language for formatting reports from existing PHACT databases. Available for UNIX, MS-DOS and VMS.

Price:

PHACT ISAM	$250
PHACT-rg	165
Source Code available, call for terms.	

For more information on these and other UNIX software products, call or write: UniPress Software, Inc., 2025 Lincoln Hwy., Edison, NJ 08817. Telephone: (201) 985-8000. Order Desk: (800) 222-0550 (Outside NJ). Telex: 709418. Japanese Distributor: Softec 0480 (85) 6565. European Distributor: Modulator SA (031) 59 22 22.

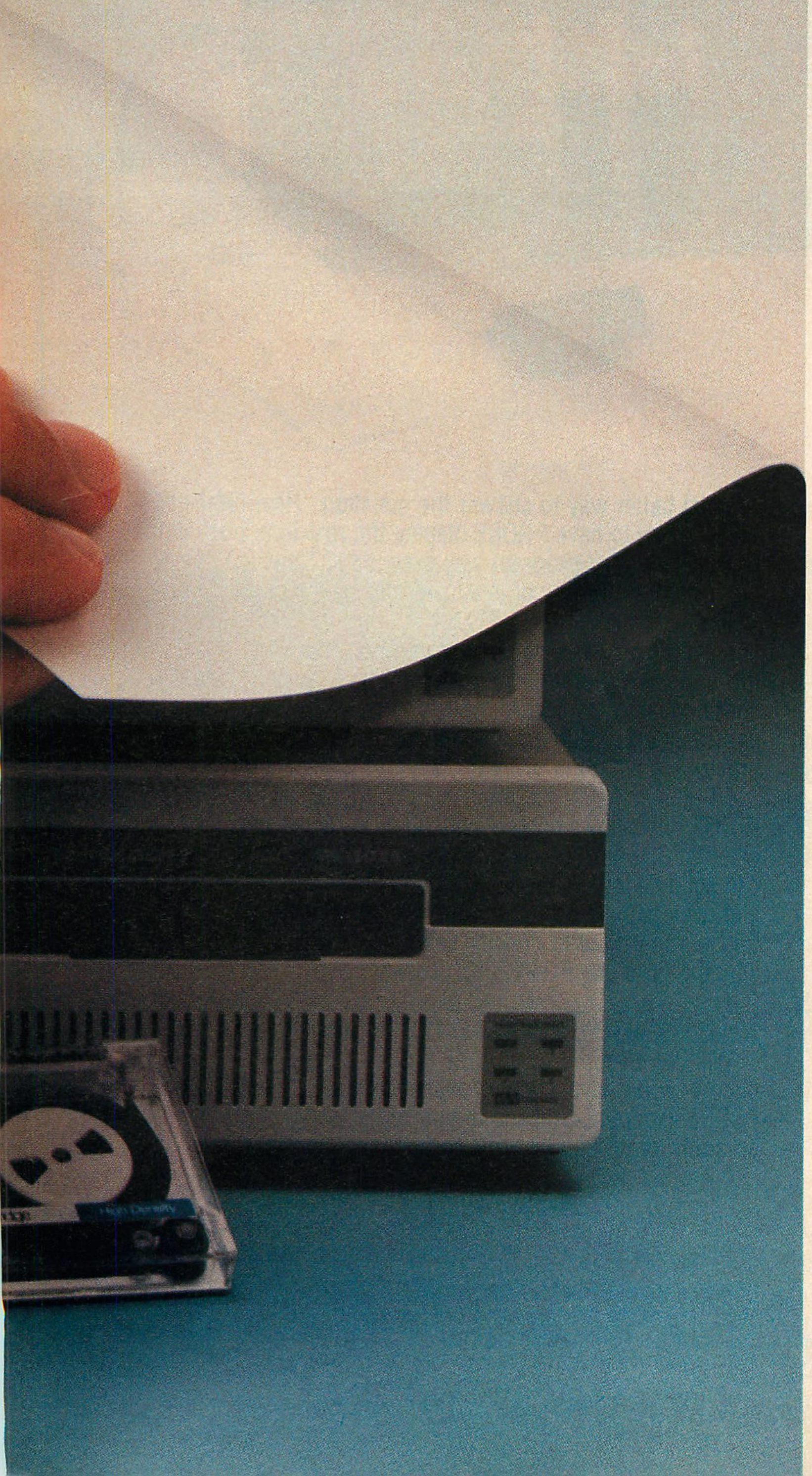
OEM terms available.
Mastercard/Visa accepted.

UniPress Software
Your Leading Source for UNIX Software.

CIRCLE NO. 199 ON READER SERVICE CARD


WE INTERRUPT
PRODUCT
BRING
INTRODUCING 5
TALLGRASS TECHNOLOGIES
TG-6135
TALLGRASS TECHNOLOGIES
Scotch

THIS MAJOR INTRODUCTION TO YOU SOMETHING REALLY IMPORTANT.



THE BIG NEWS IN DISK/TAPE IS PC/T, A SENSIBLE NEW APPROACH TO ARCHIVAL STORAGE.

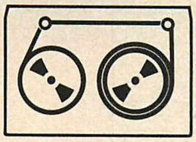
We've tamed tape. And made it docile. By making it DOS-like.

So, while this started as an ad for our five new HardFile™ subsystems, which deliver 25 to 80 megabytes of hard disk storage and 60 megabytes of tape backup, instead we want to introduce you to PC/T.™

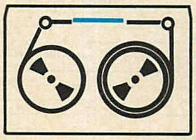
PC/T is a new format that makes tape a more sensible storage solution for personal computers. It puts tape on line, in real time, for instant access. And frees your hard disk for your most current data.

You already know how to use PC/T. Because it responds to standard DOS commands.

Here's the big news: just like any DOS-controlled hard or floppy disk, PC/T enables you to create directories and files on tape. Then you can call up the exact file you need, and change a portion of the tape without having to erase and overwrite the entire cartridge.



Without PC/T, you have to rewrite the entire 60 MB cartridge each time you make a single change.



With PC/T, you create and directly access files on tape, just like with floppy or hard disk.

PC/T formats each new tape cartridge, just like you format any hard or floppy disk, locking out bad blocks to assure that every bit of data you write to tape is recorded with utter accuracy.

What's more, PC/T gives tape true error correction capability. 50% redundancy during write operations ensures 100% reconstruction of data lost because of operator error, dust and dirt, or everyday wear and tear.

There is a catch. You can get your hands on PC/T just one way: Buy one of our powerful new HardFile subsystems. With disk plus tape. Or tape alone. Which brings us back to where we started. And gives you a place to start. Just call 1-800-228-DISK for the Tallgrass dealer nearest you.

TALLGRASS SELLS MORE HARD DISK STORAGE WITH CARTRIDGE TAPE BACKUP THAN ANYONE IN THE WORLD.

CIRCLE NO. 247 ON READER SERVICE CARD

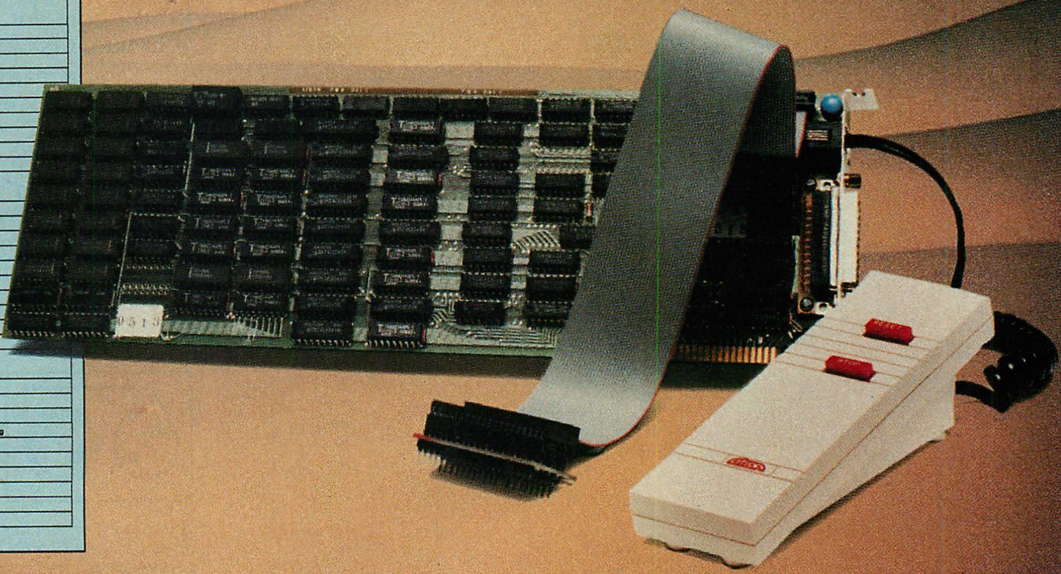
TALLGRASS® TECHNOLOGIES

COMMITTED TO MEMORY

HOW DO THE TOP-RANKED SOFTWARE FIRMS MAKE THEIR BUCKETS OF SAND SO ROCK SOLID? ASK ATRON, THE PEOPLE WHO INVENTED THEIR DEBUGGING TOOLS.

PC Week/Softsel Top List—December

		BUSINESS	
✓	1	1-2-3	Lotus
✓	2	SIDEKICK	Borland International
✓	3	SYMPHONY	Lotus
	4	DBASE III	Ashton-Tate
	5	MULTIMATE	Multimate International
✓	6	PFS:WRITE	Software Publishing
	7	TAX PREPARER 1985	Howardsoft
✓	8	PFS:FILE	Software Publishing
	9	WORD PERFECT	Satellite Software
✓	10	MICROSOFT WORD	Microsoft
		HARDWARE	
✓	1	SIXPAKPLUS	AST Research
	2	HERCULES GRAPHIC CARD	Hercules
	3	SMARTMODEM 1200B	Hayes
	4	SMARTMODEM 1200	Hayes
	5	HERCULES COLOR CARD	Hercules
	6	EXPANDED QUADBOARD	Quadram
	7	MASTERPIECE	Kensington Microware
✓	8	PC MOUSE	Mouse Systems
	9	MODULAR GRAPHICS CARD	Paradise Systems
✓	10	MOUSE	Microsoft
		SYSTEMS & UTILITIES	
✓	1	PC MOUSE/PC PAINT BUNDLE	Mouse Systems
✓	2	TURBO PASCAL	Borland International
	3	CROSSTALK XVI	Microstuf
	4	NORTON UTILITIES	Peter Norton Computing
	5	SIDEWAYS	Funk Software
✓	6	PROKEY	Rosesoft
	7	PRINTWORKS	SoftStyle
	8	SMARTCOM II	Hayes
✓	9	TURBO TOOLBOX	Borland International
	10	IN-SEARCH	Menlo Corp.



You can't pick up a PC-related magazine without seeing an editorial reference to Borland International's SIDEKICK™ program. It's one of the most popular programs ever sold to PC and PC-compatible users.

FORTUNE magazine talks about Phillip Khan's brilliantly well-executed marketing plan and a well-designed piece of software. All true. But how successful would Borland be if their product weren't absolutely rock solid?

Someone said that if your only tool is a hammer, you tend to treat all problems as if they were nails. While it's certainly possible to hammer out PC products with mere software debuggers, thousands of professionals in hundreds of companies are using Atron debugging tools.

As you can see, thirteen of the top thirty software packages were created by Atron customers. Their hammer: ATRON'S PC PROBE™, a hardware-assisted software debugging tool that plugs into your PC or compatible.

HOW THE PROS DEBUG THEIR SAND CASTLES

The PC PROBE has an umbilical that replaces your PC processor chip. Now that you can trap and catalog every signal and memory reference, you're ready to pound sand like the big boys do.

PC PROBE can trap a bug which overwrites a memory location. Because complex pointers are normally used for memory addressing, this bug occurs frequently and is quite difficult to find. As for mere software debuggers, what good is a debugger that can be wiped out by a program bug?

What better way to answer the question "How-did-I-get-here?" than to have PROBE display the real-time trace data it saved while your program was executing? Now you can do a grain-by-grain review of any sand storm.

And with the SOURCE PROBE™ upgrade to PC PROBE, you can display source-code statements* as you step through your source program. Through a window, you can continuously view critical high-level data structures. The next several source code statements are also displayed, previewing what the program is about to do.

For screen/keyboard-intensive debugging, PC PROBE's RS232 port allows screen-switching or debugging from an external CRT. For interface/hardware debugging, PC PROBE can act as a logic analyzer, trapping interrupts, I/O locations, and even signals brought in from external logic probes. All this is impossible to do with software debuggers. Which is why the pros use PC PROBE.

TODAY THE DESERT, TOMORROW THE WORLD?

Will you become king of the top-ten sand pile? Well now, history *does* have a way of repeating itself. Get your PC PROBE Federal Expressed to you today, so you can hammer on some sand, and on your competition, tomorrow.

Atron
THE DEBUGGER COMPANY

20665 Fourth Street • Saratoga, CA 95070 • (408)741-5900

*PC PROBE is compatible with Microsoft C, Lattice C, C Innovations C, Microsoft Pascal, and Assembly language. SOURCE FOR CHART: PC WEEK/SOFTSEL TOP LIST, PC WEEK, 1/29/85. Multibus™/INTEL™ Intel Corp., SIDEKICK™ Borland Interntional, PC PROBE™/MBUS PROBE™ Atron Corp. Atron also has a Multibus™ debugging product for your integrated 8086/8087/8088 applications, and the user interface is identical to PC PROBE's. Advertising by The Rick Bennett Agency, 408/258-2708.

CIRCLE NO. 203 ON READER SERVICE CARD

To Support or Not to Support?

Vendors (such as IBM) need to take a more responsive and responsible attitude

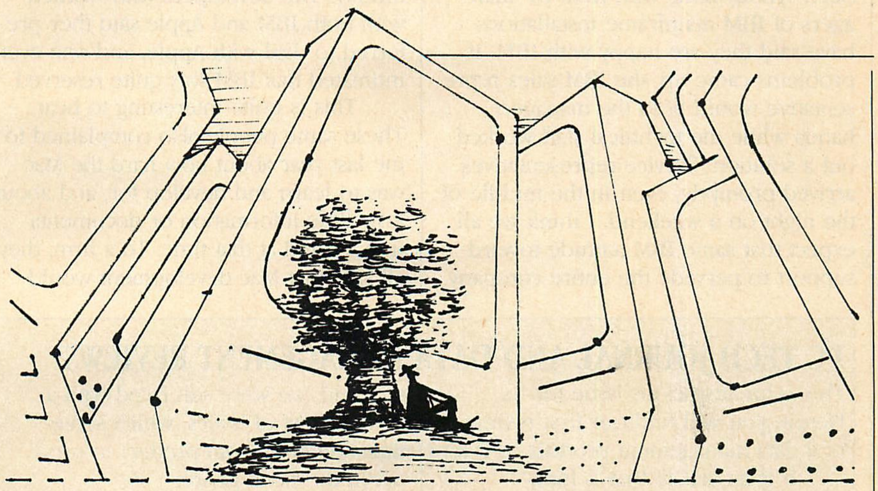


ILLUSTRATION • MACIEK ALBRECHT

In the relatively short time that IBM has been marketing the PC family, it has received many kudos for the level of support it has provided to dealers, developers, and end users. In particular, IBM has provided extensive documentation on almost every aspect of the computers and seems to have been forthcoming with deep technical information. IBM's manuals have been oft-praised for their scope, production quality, and accuracy. A significant advantage is that most documents relating to a product have been available at the time of the product's introduction.

My attitude about IBM's support is changing. Signs of serious trouble and, more importantly, developer frustration are emerging. Here are a few examples:

- IBM's update service for manuals and software is agonizingly slow. *PC Tech Journal* has not received its *Technical Reference* updates (supposedly automatic, once registered), its DOS 3.0 to 3.1 upgrades (confusion on the part of the dealer), its MASM upgrades, or its BASIC 2.0 to 3.0 manual updates. Neither has Will Fastie, private citizen, nor many others to whom I have spoken about this problem.
- Reports are circulating that newly purchased (late model) PCs have arrived with the '78 8088 copyright, the one that is marginal for use with the 8087 and has the MOV stack segment bug (see "Tracing a Bug in the 8088," Tech Notebook 4, Will Fastie, *PC Tech Journal*, September/October 1983, p. 106). If true, it represents a serious flaw in IBM's quality control.
- The 80286 now shipping in the AT has the old MOV stack segment bug. That may indicate that Intel has some problems building chips, and that IBM has accepted certain flaws as features, a fact that is sure to establish the flawed part as the standard. Having a proper chip in a machine supposedly intended for multiuser and multitasking operation would seem to be a priority on the part of the manufacturer; has IBM dropped the ball here?
- The worst case may be the Enhanced Graphics Adapter (EGA), which has been reported by many developers to have serious bugs. The documentation is said not to match the functionality of the board, and IBM apparently is not changing either one to match the other. This is particularly troubling, because the developer gets caught: is the software to be written for the way the EGA is or the way it is supposed to be? How do we tell?
- IBM recently has been shipping ATs in quantity after quite a lengthy lull. The new machines apparently have an electrical change that affects the way in which 80287 detection is handled, making previous software strategies incorrect. IBM, to the best of my knowledge, did not notify dealers or developers about this change. This is peculiar; does IBM somehow expect us *not* to figure out this change?

Let's give IBM due credit. The PC and its family members hit a resonant chord, especially in American business. But what makes the family such an unqualified success is the huge volume of software and hardware products from other sources that so greatly enhance the machine. If we had been stuck with only IBM-label add-ins and add-ons, we most surely would not have improved our productivity to such an extent, and it is unlikely that we would have bought quite so many PCs.

The problems mentioned above are irritants to the end-user community, but they are disasters waiting to happen for developers. It should not be the case that a developer receives no information from the vendor after independently finding a problem and reporting it. It should not even be the case that a developer has to find the problem alone; IBM should let developers know about these bugs. Fixing the bugs is at IBM's discretion, but getting the information out is something a manufacturer must not fail to do.

Recent discussions with a number of developers confirm that support from IBM is limited or nonexistent, and that anger is brewing. Some companies have the technical expertise to research such problems and identify a solution, but most firms, especially smaller shops, do not. Tracking down an obscure bug in the firmware or discovering a flaw in the processor is a duty we all can legitimately expect the vendor to do.

The most obvious evidence that IBM is having trouble delivering a reasonable level of support is the *Technical Reference* manual. With just one computer and a few options, a single

book was enough—and highly treasured at that. As the number of machines and options increased, IBM added a new book for each machine and later a separate, two-volume set for options. This looks good, but IBM has not delivered the updates, especially the much sought after EGA technical information. At first IBM could not even say how a developer could obtain the EGA documents; dealers had no idea at all.

Support and service have always been synonymous with IBM. DP managers of IBM mainframe installations have said they are happy with IBM. If problems came up, the IBM sales representative would hold the managers' hands while the technical staff worked out a solution. Service representatives arrived promptly, even in the middle of the night on a weekend. I think we all expect that same IBM attitude toward support to pervade the entire company. Without question, dealers and developers should be able to get more support than they do now.

By the way, I recently conducted an informal poll of a few Apple II and Macintosh developers. They all said that Apple had its problems with developer support, but at the same time they gave Apple high overall marks, pointing out that the company seemed to have a genuine interest in and a sense of responsibility toward its developer community. The developers who worked with both IBM and Apple said they preferred to deal with Apple, and one even intimated that IBM was quite reserved.

This is quite interesting to hear. These same people also complained to me last year about how hard the Mac was to learn and develop for, and about how little information or documentation existed at that time. To a firm, they all said that Mac development would take 50 to 100 percent longer than the amount of time it would have taken on a conventional machine, such as the PC. Even after the initial learning curve, most developers predicted that future products would be as much as 25 percent more expensive to develop on the Mac. Today, the developer's attitude toward Apple seems to be very positive, a state of affairs very much in evidence during the heyday of the Apple II+; apparently, this part of Apple's corporate culture still exists.

Based on the remarks of the developers to whom I have spoken, IBM should be cultivating a more meaningful relationship with outside product sources. With several actively marketed machines, more anticipated, and a soft market, now is an ideal time for IBM to strengthen these relationships to ensure that the vital flow of innovative and creative products continues.

PC TECH JOURNAL AND DATA MANAGEMENT REVIEWS

This is an auspicious issue for us. Herein you will find our first review of a data management product.

It is an article that is long-awaited. We have been looking at these products since we began the magazine more than two years ago, and we have been working during that time to decide how best to present these products to our readers. Perhaps you will say that our approach should have been obvious; I only wish it had been.

Confusing the issue are the many fine products that our readers might find useful, but that did not seem to be the kind that we would be expected to review. Once we narrowed the field, we were still faced with a complex set of issues, which sometimes differed from product to product in the same genre.

Executive editor Julie Anderson put a lot of work into this project. She examined innumerable products and visited any number of booths at trade shows that even hinted at data management. She pulled out her old textbooks and boned up. She beat on me for better guidance and the rest of the staff for ideas and, in the end, produced the specifications that are presented in the introductory article on data management (page 46).

Julie will be supervising three or so of these reviews at any given time, and our specifications will surely undergo refinement and improvement. Our first author (or was it guinea pig?), Chris Christian, provided many insights for which we are especially grateful, and the other authors with work in progress also have been forthcoming with suggestions.

The reviews of data management products, which we consider of prime importance, will make up a substantial portion of *PC Tech Journal* in the foreseeable future. I urge you to consider these articles carefully and let us know what you think. Your thoughtful suggestions can only improve our work and make your investment in *PC Tech Journal* ever more valuable.

PCTECHline ONLINE!

Thanks to the efforts of technical editor Jeff Duntemann, who researched BBS software and painstakingly crafted our online system, PCTECHline is now operational. The number is 301/576-PCTJ. Except for the time needed to service the system, PCTECHline will operate 24 hours a day, every day.

Currently, the bulletin board software we are using is the BBS-PC package from Micro-Systems Software, Inc., of Boca Raton, Florida. Baud rates supported include 300, 1200, and 2400. Download protocols include straight ASCII file; XMODEM checksum; XFER, a CRC protocol by Micro-Systems; and Ultraterm, a protocol by United Software Associates.

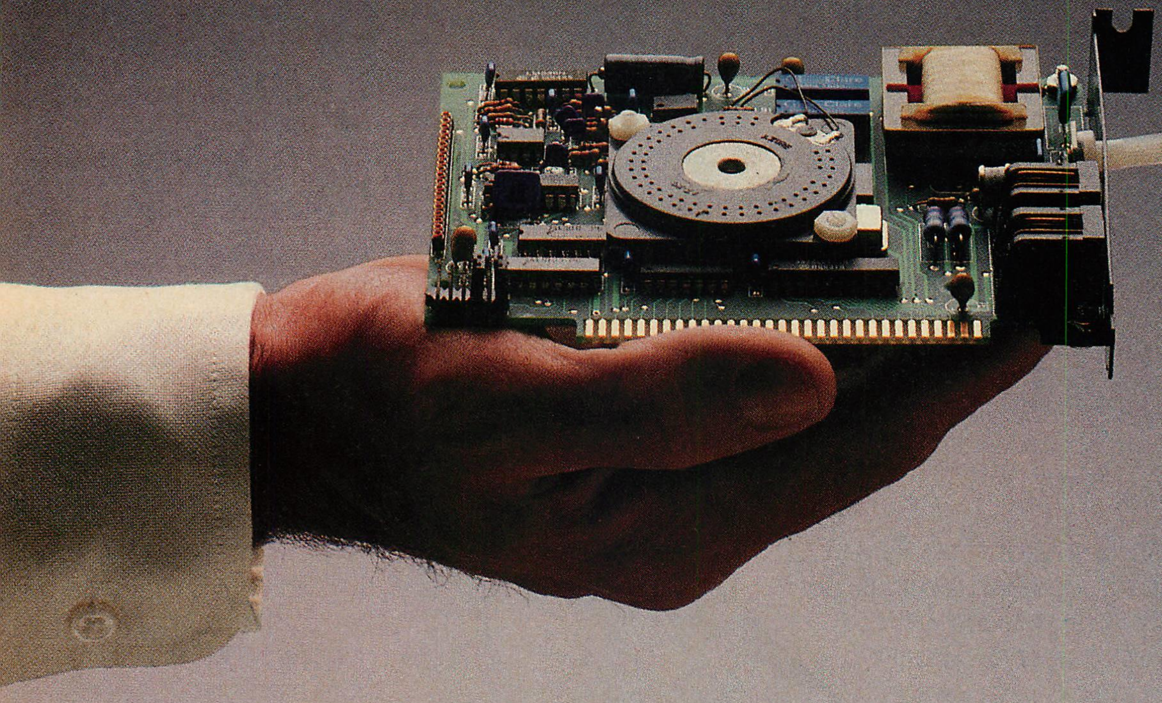
The primary purpose for PCTECHline will be to make article listing files available immediately for downloading. Later, we will have a downloadable cumulative index in one or more popular database formats and straight ASCII. The message board will be disabled except to allow comments to Jeff, who will act as sysop. Any comments we receive will be considered as publishable letters to the editor, unless the sender explicitly states otherwise. All comments will become the property of Ziff-Davis Publishing Company.

In the beginning, we will operate only a single telephone line. We have no idea what your response to the service will be, so we have no way of knowing if a single line is enough. *Please* let us know if you have trouble getting through by dropping us a note (US Mail, MCIMail) or by calling Jeff at 301/576-0770.

We have tried to create a smooth system, but we expect a few wrinkles at first. As we all gain experience, the system will be adjusted as required. If we find better software, we may even change the system.

Again, the number is 301/576-PCTJ. Let's see how it goes.

QUADRAM ENGINEERED A BETTER-THAN-HAYES MODEM...



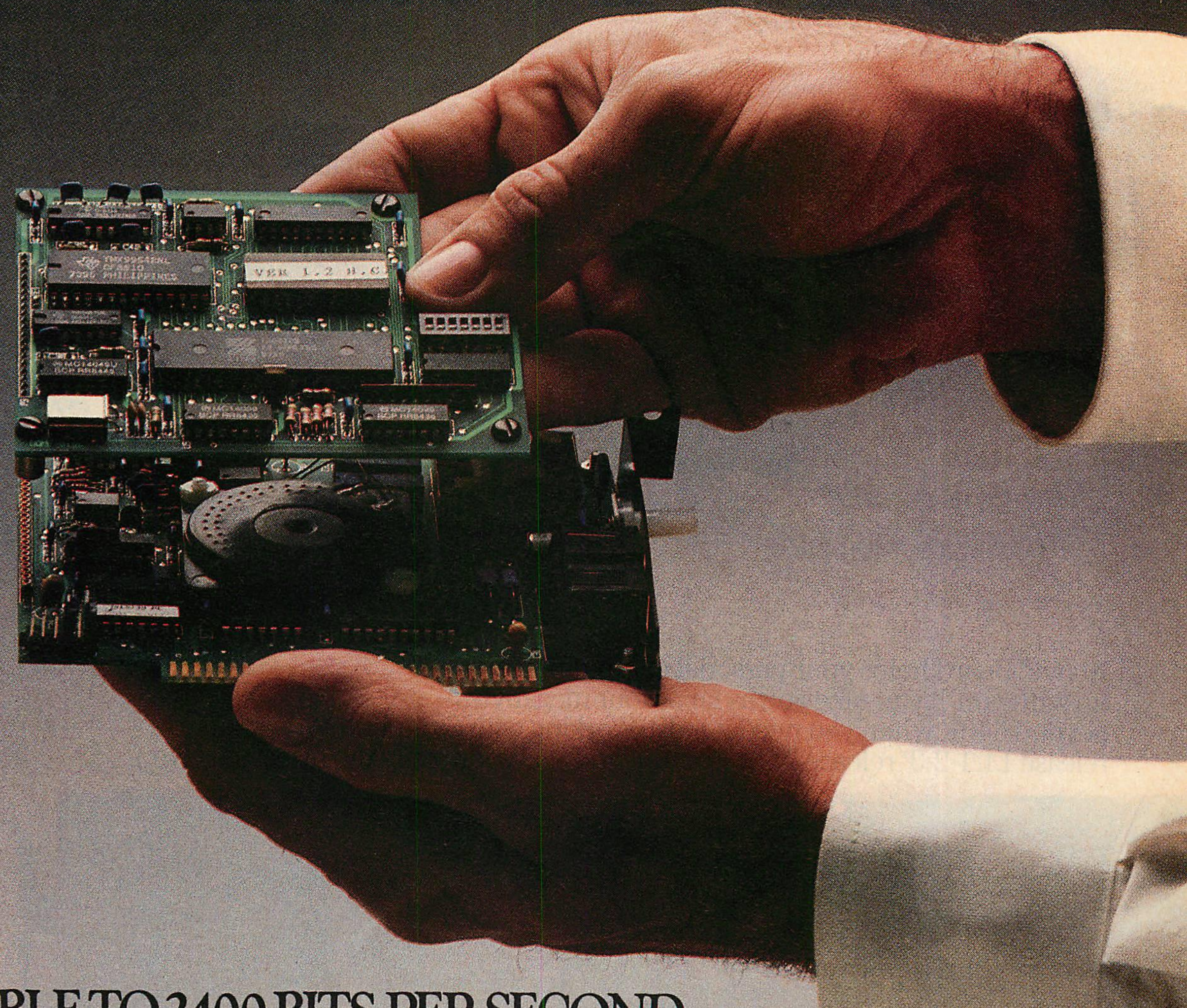
INTRODUCING QUADMODEM II

New from Quadram, a 300/1200 bits-per-second modem that meets all the standards and then tops them with a format that grows with tomorrow's technology. QuadModem II gives you everything you'd expect from your communications investment. And more.

Like full-featured performance on an internal half-card. QuadModem II fits comfortably in the short slot on your PC/XT/AT/Portable or compatible. It comes complete with powerful CrossTalk XVI communications software. And commands compatible with all the right standards worldwide. So easy to use you'll be up and talking in no time.

QuadModem II	Hayes
☑ 300/1200 bps. Upgradable to 2400 bps.	☐ *Buy a new modem.*
☑ 2400 bps on an internal half-card.	☐ *External modem only. Bulky, messy cabling, takes up desk space.*
☑ Half-card size fits in PC/XT/AT/Portable short slot.	☐ *Full-card wastes space & reduces flexibility.*
☑ CrossTalk XVI software at no extra charge.	☐ *Smartcom II $95 extra.*
☑ Nationwide Product Support Centers.	☐ *None provided.*
☑ Compatible with all communications standards.	☑ *Yes.*
☑ Part of an integrated family of communications products.	☐ *No.*

AND THEN TOPPED IT.



UPGRADABLE TO 2400 BITS-PER-SECOND

As standards evolve, you'll be ready with the QuadModem II 2400bps upgrade kit. Just snap it on the existing board and get high-speed communications in a short-card slot! QuadModem II lowers your dial-up costs and still communicates with lower-speed modems. Automatically.

Insist on the new standard... QuadModem II. Part of an integrated communications family engineered for excellence and 100% PC tested.

To find out more, visit the dealer nearest you. Or contact us at 1009 Mansell Rd., Roswell Georgia 30076; 404-993-4590. **Or call us toll free 1-800-334-9339.**

SPECIFICATIONS

Hardware
Internal half-card, Bell 103/212A/Hayes compatible

System Compatibility
IBM PC/PC/XT/AT/Portable and compatibles

Communications Speeds
300/1200bps, upgradable to 2400bps

Communications Modes
Full/half duplex

Dial States
Pulse (Rotary)/Touch Tone

Call/Answer States
Manual Originate/Auto-answer/Auto-dial

Call Progress Monitoring
Complete visual advisory of communications progress

Operating Features
VT52/100/102 emulations, data buffers and verification, character string searches, operates from hard disk.

QUADRAM™
An Intelligent Systems Company

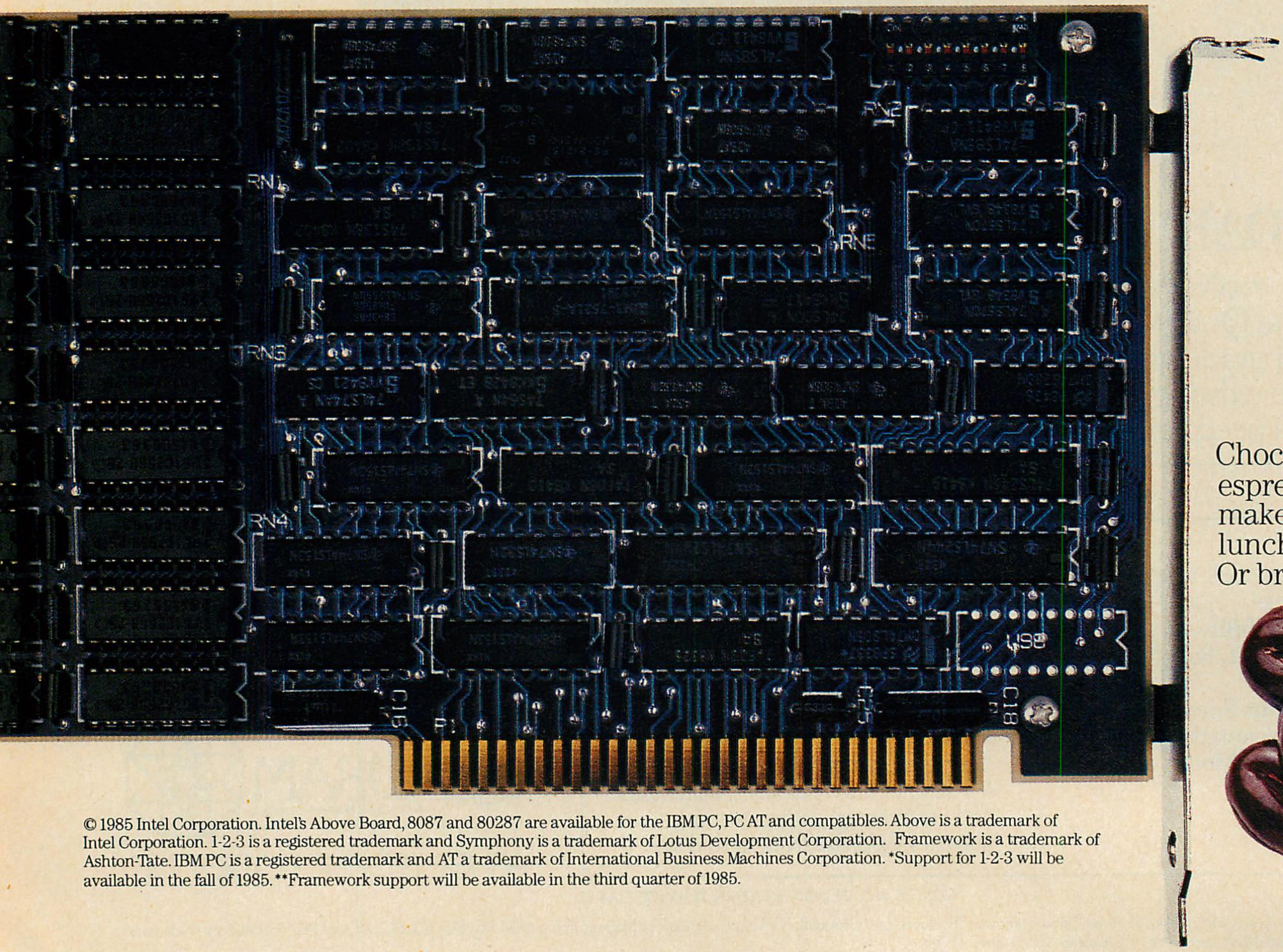
ASHER TECHNOLOGIES INC.

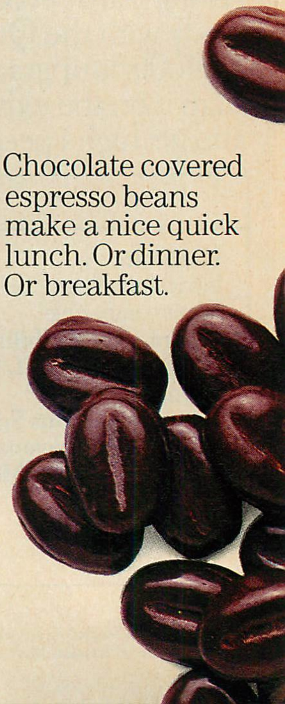
CIRCLE NO. 211 ON READER SERVICE CARD

Survival kit for

Jumping rope helps keep you alert when you've put in a 6 hour day before noon.

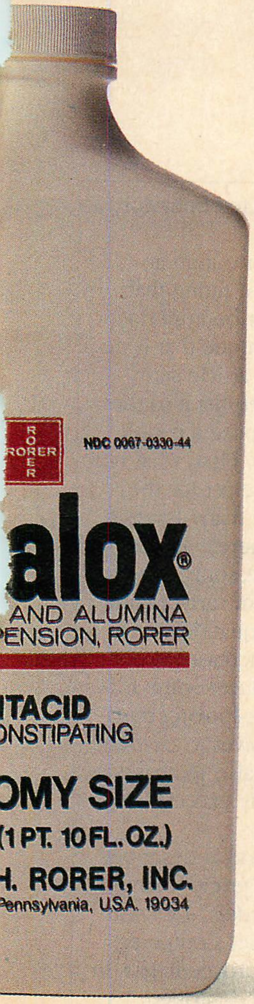
Above™ Board helps your personal computer keep up with you. It can take you as far as four megabytes beyond 640K when you need it most: Working with Symphony™ release 1.1, 1-2-3® release 2,* or Framework.™** (Where only the fittest survive.)





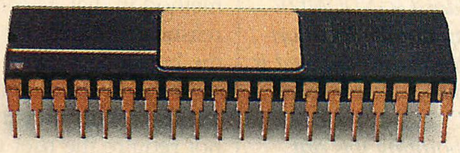
Chocolate covered espresso beans make a nice quick lunch. Or dinner. Or breakfast.

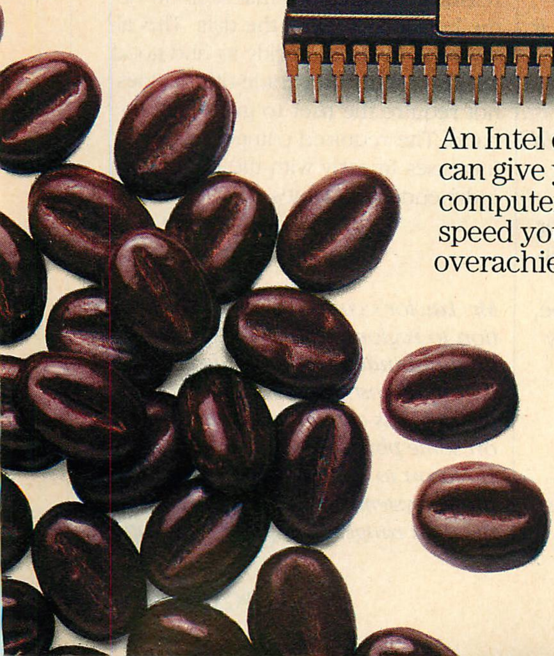
overachievers.



Two old sweaters for when you're working late and you can see your breath because they've turned off the heat in the office.

When you get into Q4 on your spreadsheet and run out of memory, Maalox can help soothe your stomach.





An Intel coprocessor can give your personal computer the extra speed you need to keep overachieving.

Life isn't easy when you're overworked and underappreciated. That's why Intel created Above Board. It's a new kind of memory board designed to meet the specification jointly developed by Intel and Lotus Development Corporation.

Above Board does all the things you'd expect any self-respecting memory board to do. But because it lets your personal computer work so much harder, you can put off such uncomfortable things as cutting and pasting. Or such disruptive things as converting up to a mini.

You can also get an Intel coprocessor to give you the extra speed you need to keep up with all that new memory. Intel has coprocessors that will let your IBM PC® or PC AT™ run up to five times faster. Or more.

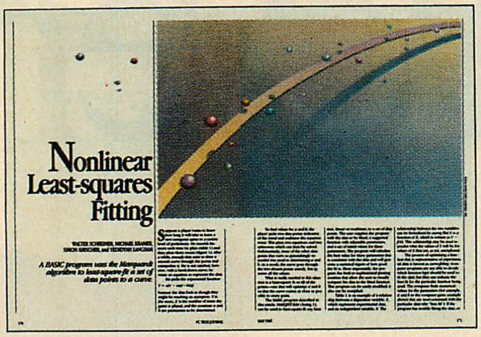
You can see and touch and buy these Intel products at your favorite computer store. Or call 800-538-3373 for the names of dealers near you.

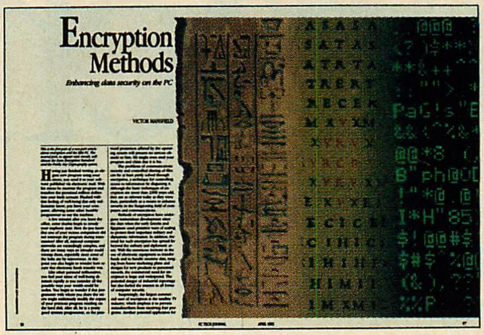
And find out how to keep your personal computer working as hard as you do.

Help for the overachiever.

intel®

CIRCLE NO. 216 ON READER SERVICE CARD





RANDOM QUESTIONS

A hearty "hats off" on your quality publication. Your staff seems to be reading my mind in publishing the articles I need and when I need them. The B-tree series ("Tree Structures," Atindra Chaturvedi, February 1985, p. 78; March 1985, p. 131) came at the precise time I was writing an order form writer for my own business. "Encryption Methods" (Victor Mansfield, April 1985, p. 96) also came at a good time.

I noticed in the KEYGEN.PAS listing that Mr. Mansfield used the Turbo predefined procedure RANDOMIZE. On page 20-2 of the *Turbo Tutor*, Borland makes the confession that "... the procedure RANDOMIZE does nothing at all in Turbo." This confession, however, is immediately followed by the listing of a procedure to take the place of the "do nothing" procedure.

Apart from technical correctness, the only other reason to use the substitute procedure is because implementing the original RANDOMIZE procedure may lead to the same keys being generated over and over again.

From the program comments in the *Turbo Tutor*, it would seem that the whole purpose of the RANDOMIZE procedure is to store a 32-bit seed starting at memory location D segment, offset 129H. The new RANDOMIZE procedure gets this seed from the DOS TIME function if the clock is ticking (another procedure if it isn't).

Reading between the lines, I deduce the old RANDOMIZE procedure does not get this seed to the memory location mentioned above. This means that the seed is invariable 0 when the computer is powered up. My understanding of pseudo-random generators is limited, but I did learn that pseudo-random generators generate identical sequences when initialized with an invariant seed.

Eugene H. Laird
Vallejo, CA

After the program KEYGEN.PAS was published I, too, discovered the startling admission in the Turbo Tutor *that the RANDOMIZE function "did nothing." However, your fear that the same key would always be generated without a working RANDOMIZE function is ungrounded. I have run several chi-square tests on the random number function in Turbo, versions 1.0, 2.0, and 3.0, without using any RANDOMIZE function (neither the old incorrect one nor the fix offered in the* Tutor). *A cursory glance shows that without RANDOMIZE the same sequence of numbers is not generated by repeated calls to RANDOM in any version of Turbo Pascal. A correctly functioning RANDOMIZE should bring the pseudo-random number sequences closer to a true random number sequence; this was not immediately evident in my quick tests, however. For added security it would be wise to add the suggested fix that is offered in the* Turbo Tutor, *but the probability of generating duplicate keys without RANDOMIZE is quite small.*

Borland told me that the problem has been fixed in Turbo 3.0, but repeatedly calling the standard RANDOMIZE function and reading out the location where the Turbo Tutor *claims the seed is held showed the number at these locations never changed.*

Thank you for pointing this out.
—Victor Mansfield

REASONABLE EQUATIONS

My thanks to Walter Schreiner, et al, for the article, "Nonlinear Least-squares Fitting" (May 1985, p. 170). The underlying methodology is difficult to describe, but curve fitting can be useful in many areas for those who take the time to learn to use it properly.

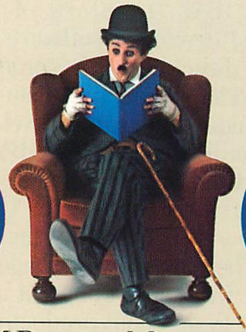
A word of caution, however. The use of curve fitting to project beyond the data (as in the author's leading example) is risky. Many functions can be "forced" to fit within the range of the data, but exhibit unexpected behavior outside the range.

I would like to point out an approach to nonlinear curve fitting that has a property that might be important to some users. The technique that is described in the Schreiner article starts with an equation and then fits it to the observed data. This can pose a couple of problems for the user. First, what is a reasonable equation to select for the observed data (there are many issues and trade-offs involved here). Second, if there are a number of reasonable equations for one independent variable, the number of reasonable equations for *N* independent variables increases exponentially with *N* and produces some formidable and nonintuitive-looking equations. The computing power to examine all of these possibilities also increases exponentially and quickly exceeds supercomputer capability.

An alternative method is described in "Techniques for the Development of Empirical Subsystem Models" by Racite and Lawlor (1972 Summer Computer Simulation Conference proceedings). In short, it differs from the described method in that it "constructs" a set of reasonable equations that seem to do the best job of fitting the data. This allows the method quickly to find good fits that reflect user trade-offs. It does not require the user to guess beforehand. The required computing power increases linearly with the number of independent variables.

Frank Lawlor
Saugerties, NY

Mr. Lawlor's concerns about extrapolation to regions beyond the measured data are indeed well-founded. Extrapolation, almost by definition, implies that the user has chosen to fit the data based on some physical model, and thus there is reason to ascribe to the system a continued systematic behavior outside of the data range. Attempting to extrapo-

READ  ONLY

A review of the IBM Personal Computer Family. Vol. 2 No. 2

WHAT'S THE PROGRAM?

Meaningful dialogue. There are two programs from IBM that can greatly improve the quality of our running dialogue with computers. They are the IBM EZ-VU Runtime Facility and the IBM EZ-VU Development Facility.

Think of the EZ-VU Runtime Facility as a mediator in your IBM PC conversations. It handles the exchange of commands and information between you and your application programs through predefined screens. It can give you a single consistent interface with applications written in a variety of languages.

In short, the EZ-VU Runtime Facility lets you concentrate on the essentials of the job you're doing.

If your job is program development, the IBM EZ-VU Development Facility can help you write menu-driven applications—or revise existing ones—that are both sophisticated and easy to use. It incorporates a screen design tool that works through the function keys on your IBM Personal Computer, so there are no special codes and commands to slow down your design work. EZ-VU also helps make fast work of testing and revising your screen designs.

Two additional points. Both of these EZ-VU programs benefit from years of success by similar dialogue management programs in IBM host MVS,VM, and VSE operating environments. And both run under the IBM TopView program, which allows you to run a number of software applications concurrently.

A quantum leap. Speaking of technological advances, IBM Personal Computer Professional FORTRAN represents a quantum leap forward in FORTRAN for microcomputers.

It's a full ANSI 77 implementation with enhancements that offers an un-

Application management, program development, and communications software from IBM. See next page for IBM PC Network SNA 3270 Emulation Program story.

usual combination of speed and accuracy. Optimization techniques and features such as a full symbolic interactive debug facility are similar to those usually found only in IBM VS FORTRAN and other mainframe FORTRAN compilers.

You can use IBM PC Professional FORTRAN to work on large or small host programs and to recompile existing FORTRAN programs—or sections of those programs—to run on your IBM Personal Computer. It's ability to handle arrays larger than 64KB gives you the equivalent of mainframe capability on a personal computer.

And IBM PC Professional FORTRAN was designed for IBM by Ryan-McFarland Corporation to help you take full advantage of other IBM Personal Computer software, such as the IBM Personal Computer Engineering/Scientific Series graphics development tools.

Make that "quantum leaps."

HARDCOPY

Hidden talent. Think of the many entertaining and useful programming ideas that must exist out there but never find their way to market.

The IBM *Directory* of personally developed software gives you direct access to some of that hidden talent. It's a catalog of unique programs developed by individuals for the IBM Personal Computer Family.

Programs listed in the *Directory* sell for as little as $14.95. They cover a wide range of interests, from entertainment and education to personal productivity and business applications.

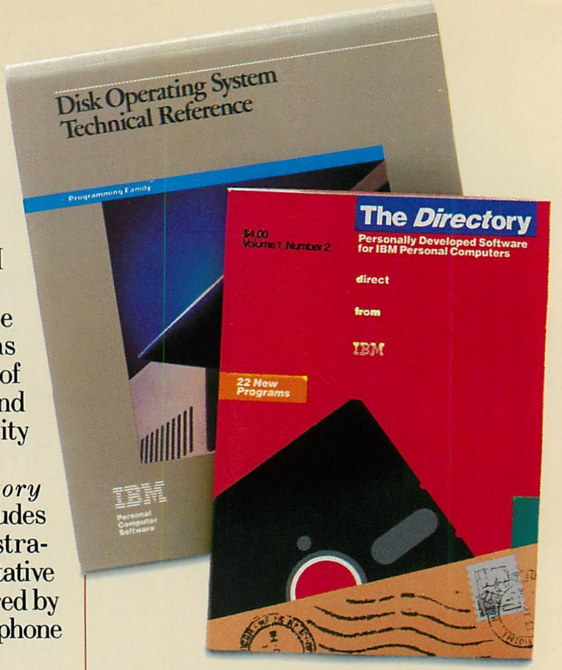
Each program in the *Directory* has a full description that includes system requirements and illustrations or color photos of representative screens. Programs may be ordered by mail or through a toll-free telephone number.

To subscribe to the *Directory*, call 800-IBM-PCSW.

The last word. Or perhaps we should say the last word to date. The new IBM DOS Technical Reference manual contains just about everything you'd want to know about the IBM Disk Operating System Version 3.1 and previous versions 2.1 and 3.0.

That's not to say that new improvements and information won't appear in the future. They will, and you'll be kept abreast of such developments. An update information service is included in the manual's purchase price.

So you'll always have the last word.



Updated versions of the DOS Technical Reference manual and the Directory of Personally Developed Software from IBM.

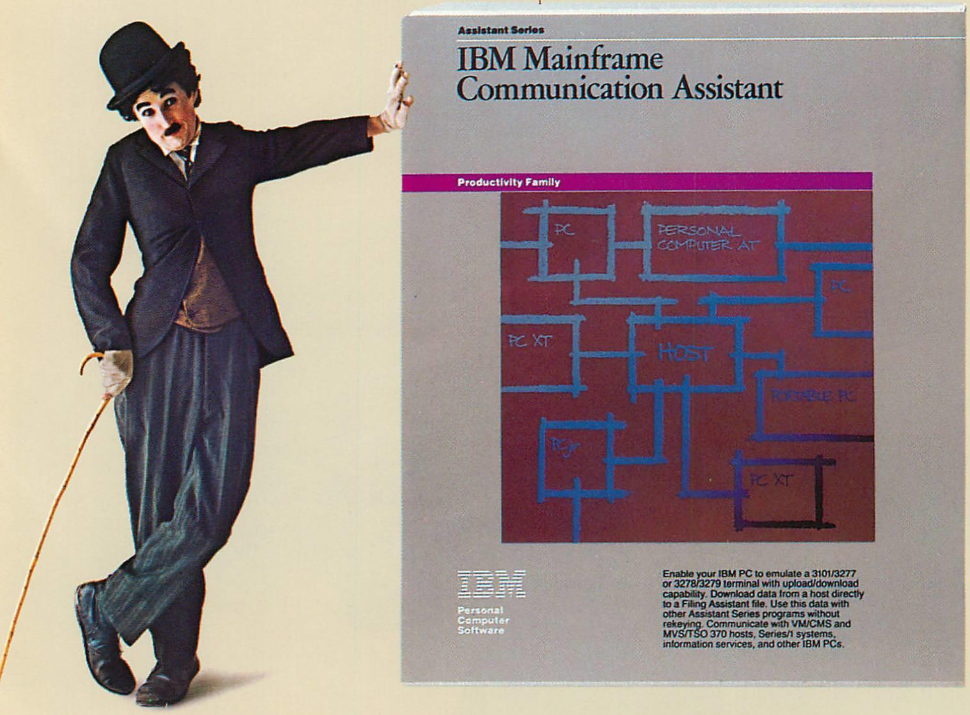
FAMILY TIES

Close connections. The IBM PC Family has always included hardware and software to help you keep in close touch with important contacts.

Last year's announcement of the IBM PC Network, for instance, was an important milestone in communications among the immediate family. It gives you an easy way to share information and hardware resources like printers and disk storage devices.

There are two recent IBM communications software products that extend those IBM PC Family connections even further. They make it possible for you to work directly with data stored on an IBM host computer, to communicate from one network to another, and to do an even wider range of your daily business over the IBM PC Network.

Host communications. The IBM PC Network SNA 3270 Emulation Program, for example, allows your IBM Personal Computer to communicate with an IBM host system through telecommunications lines.



Broaden your IBM PC connections with communications software from IBM.

You then have direct access to the data and programs on the host computer. So if you're working on a branch-office quarterly report, you no longer have to wait while essential data from a headquarter's computer is sent to you and copied for your use. There's also a redirector function that allows you to place the data on a file server for distribution to other stations on your IBM PC Network.

In addition, when the program is installed on an IBM Personal Computer in an IBM PC Network, that PC can act as a communications gateway for other members of the same network. If members of a network need to communicate with more than one IBM host computer—or with different applications on the same host—multiple gateways can be attached to a single IBM PC Network.

And an IBM PC with the IBM PC Network SNA 3270 Emulation Program installed can still be used as a network station doing standard PC work.

You could, for instance, use your IBM PC to create a data set on a host system. You could also create a word processing project on your IBM PC using DisplayWrite 3,* switching easily back and forth between them without terminating either session. The IBM PC Network SNA 3270 Emulation Program also allows you to transfer graphics printing jobs from the host to an IBM PC with a graphics printer attached.

IBM Quietwriter® Printer offers advanced printer technology and unusually low operating noise level. Shown with sample of letter-quality printing.

Stand-alone assistance. There's a new addition to the IBM Assistant Series to help handle stand-alone communications between your IBM Personal Computer and a host machine or another IBM Personal Computer. IBM Mainframe Communication Assistant software includes a 3101 and 3270/78/79 terminal emulator for your IBM PC, plus a set of host computer support programs for VM/CMS and MVS/TSO systems.

Mainframe Communication Assistant has the same easy menu structure as other members of the Assistant Series, such as Writing Assistant and Planning Assistant. And it offers a number of unusual features to simplify your communications work.

It can, of course, speedily transfer files between host and PC (or PC and PC). After you've worked with the host file, Mainframe Communication Assistant lets you transfer only the changes you've made—rather than the entire file—back to the host computer.

And, as a member of the Assistant Series, Mainframe Communication Assistant allows you to integrate mainframe database information into Filing Assistant files. You're then able to transfer that information to other members of the Assistant Series, saving the time and effort of reentering data that has been stored on a mainframe computer.

*Follow the proper installation instructions in the IBM PC Network 3270 Emulation Program documentation for DisplayWrite 3.

HARDWARE NEWS

Quiet, please. There's been a quiet revolution in printer technology. The IBM Quietwriter® Printer is as advanced as the computers it serves. It's versatile, compact, remarkably quiet, and produces work that might make people think you've got a printing press in the back room.

In this case, the spark behind the revolution is a new method of resistive ribbon, non-impact printing developed by IBM. The "Quietwriter" Printer replaces print elements or hammers that strike the page with a unique multi-layer ribbon and print mechanism that virtually "paints" characters on the paper.

As a result, the "Quietwriter" Printer produces superb, letter-quality printing on a variety of papers and in a wide range of type styles. To change type styles you just unplug one font module and plug in another. And because the "Quietwriter" Printer can accommodate two font modules, you can have two type styles online at once.

The "Quietwriter" Printer also allows you to produce a wide array of character graphics either separately or to highlight reports and correspondence.

And it does all of this very, very quietly. At 50 dB, the IBM "Quietwriter" Printer* makes less noise while printing than many printers make while idling. That means you can put the "Quietwriter" Printer wherever it's most convenient; it won't disturb either your telephone conversations or your train of thought.

Unconventional ability. IBM also makes conventional printer technology seem anything but conventional.

Consider the IBM Wheelprinter,* for example. Its printwheel is designed to provide sharp, clear letter-quality printing even after millions of impressions. And the printwheel is easy to change, which is important because there's a selection of over 500 printwheels to choose from. The Wheelprinter's standard features also include both automatic sheet feed and continuous forms feed—optional on many other printers.

The Wheelprinter is reliable enough to take on high-volume office work. It even works with two types of ribbons to suit the requirements of different types of jobs. There's a single-strike ribbon for finished reports or correspondence. For more routine jobs like purchase orders or internal memos, you can use a longer lasting, lower cost multi-strike ribbon.

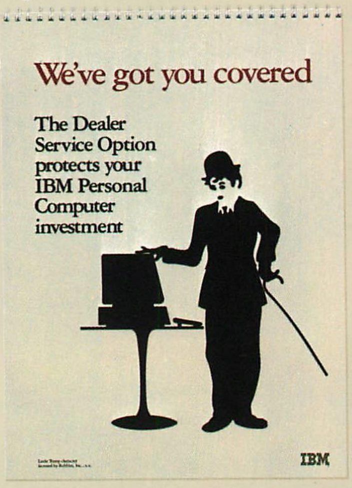
The Wheelprinter has equally impressive qualifications as a home printer. To begin with, it's remarkably easy to use. The Wheelprinter's integrated paper path provides reliable cut-sheet printing and allows it to do much faster work than you might expect from a 25cps printer. And its acoustically engineered cover makes the Wheelprinter an exceptionally quiet impact printer.

Maybe "conventional" isn't the right word at all.

*These are just two of the various printers available from IBM for the IBM Personal Computer Family.

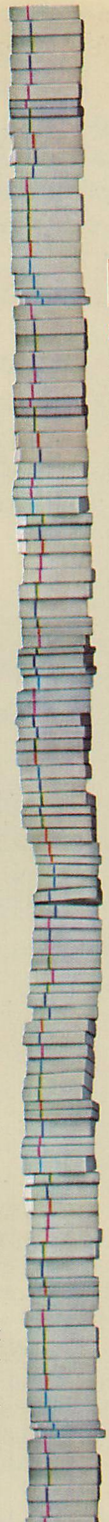
ON THE STOREFRONT

Extra attention. Even the best equipment sometimes needs a little extra attention. An IBM Dealer Service Option can provide it. It gives you extended service coverage for IBM Personal Computer products, and is available from participating Authorized IBM Personal Computer Dealers either before or after your original warranty expires.



In addition to sparing you the complications of arranging for the job to be done after something has gone wrong, a Dealer Service Option assures you of first-rate work. IBM Personal Computer Dealers have full-time, IBM-trained and authorized technicians on staff as well as direct access to IBM technical experts who can help with unusual or complex questions.

An IBM Dealer Service Option also covers a couple of important additional contingencies. If you move, your service coverage can be honored by a participating Authorized IBM Personal Computer Dealer in your new location. And if you should decide to sell your IBM Personal Computer product, your Dealer Service Option may be transferred to the new owner.



IBM®

TIPS AND TECHNIQUES

Stop action. If you occasionally sit frozen while screen after screen of information rolls by too fast to read, take heart. There's more than one way to stop that cascade of data and view one screenful at a time.

When listing the directory of a diskette or fixed disk, the command DIR/P will do the trick. After finishing with one screen, press any key to bring up the next.

To slow down the listing of a text file, you could use the CTL/NUM LOCK keys, but that involves keeping both hands on the keyboard and an eye on the screen.

Instead, check your DOS directory listing to make sure the DOS utility program MORE.COM is available. Then, at the DOS prompt A>, type the command line MORE<filename and press enter.

NOTE: be careful to use "<" and not ">"; if you enter the wrong one, you'll destroy your text file.

To view a file called PC-WRITE.DOC, for example, enter MORE<PC WRITE.DOC. That will list a single page of text on your screen and display the message —MORE—. Again, press any key to view next screen.

Voilá, perfect control.

Thanks for this tip to Chuck Harrington of the Athens, Ohio, area IBM PC Users Club.

Thanks also—and apologies for omitting a note of credit in the last issue of *Read Only*—to Ed Smuckler of the Redondo Beach, California, Greater South Bay User Group for his tip about setting screen colors.

For more information about IBM Personal Computer products discussed in this issue of *Read Only*, see your Authorized IBM Personal Computer Dealer or IBM Product Center. To learn where, call **800-447-4700**. In Alaska and Hawaii **800-447-0890**.

late without such a model is a mistake often made by less experienced users.

One topic that we did not attempt to address in the article is when the use of least-squares methods is appropriate and when other methods should be used; however, since the issue has been raised, a few comments are in order.

In general, least-squares methods are appropriate when one is modeling data that obey some physical principle or must be constrained to do so, and where the errors on the data points are all similar or can be estimated individually. If these criteria cannot be met, then one should alternatively consider trend fitting or data-smoothing methods. Lawlor's method, while employing least-squares minimization techniques, actually falls into this latter category since the fit to the data is given more importance than any specific model of the system that gave rise to the data. Two other important trend-fitting methods that we have used in the past are cubic splining (which also uses least-squares minimizations) and the less well-known data-smoothing techniques of Tukey (John W. Tukey, Exploratory Data Analysis, *Prentice-Hall, 1977).*

A better fit to the data in the example given in the article could probably be obtained using data-trend methods because a progression of game scores does not intuitively follow some physical principle. However, precisely because we wished to extrapolate *the data, it was essential to choose a function with some appropriate behavior in the asymptotic region. Trend-fitting techniques are not well-suited for making such choices. Surely, quite unexpected behavior would have resulted from extrapolation had the software algorithm "constructed a set of reasonable equations that seem to do the best job" of fitting the observed data. It is evident that both the choice of fitting function and fitting method are often constrained as much by one's objective as by the physical process giving rise to the data.*

Probably because of our physical science background, we feel that the importance of selecting an appropriate model transcends the technical task of fitting the data, even at the expense of not obtaining a perfect fit. Hence, we use trend-fitting methods very sparingly. In spite of this bias, we would love to have the Racite and Lawlor algorithm in our tool set of data-fitting techniques, and we invite them to consider publishing an IBM PC implementation.

—Walter Schreiner, Michael Kramer, Simon Krischer, and Yedidyah Langsam

RECOVERY REASSURANCE

We wish to thank Robert Sanford for his kind words concerning the BABY/34 product ("System/34 for the Microcomputer," May 1985, p. 193). However, since he chose to highlight what is described as a flaw in the recovery process, we feel that the following information will reassure the user.

Each time statements are displayed on the screen, the work files that are referred to are forced closed and reopened. This happens every time the mode is changed (ENTER, UPDATE, MOVE, INCLUDE, SCAN) and each time the Roll Up key is pressed. Since these actions are a normal part of a normal editing session, the most data that can be lost in any given session is only the data that have been changed since the time of the last screen display.

Since considerable time and effort was expended in the design and implementation of the recovery process, we would not like the user to be left with the impression that any time the power goes off, he will have to use some external tool to recover his data.

Jerry Funk
California Software Products
Santa Ana, CA

HACKER ANTIDEFAMATION LEAGUE

I am writing regarding "The Tort of Copy Protection" (Max Stul Oppenheimer, March 1985, p. 177). This otherwise great article has one major and quite offensive inaccuracy. The author refers to computer crime laws as "anti-hacker statutes," which is wrong. These laws are anti-criminal statutes and have nothing to do with hackers.

For over 20 years, *hacker* has meant somebody who knew a great deal about computers and enjoyed using them because they were fun. Recently, a new group of computer users have started calling themselves hackers. Just because that is what they call themselves does not mean that is what they are. If a bunch of plagiarists started calling themselves writers or editors, I don't think everybody would automatically believe them. This new group is best referred to as criminals since that leaves no doubt that they are no better than other lawbreakers.

I have come to expect that the mass media will misuse the term, but I expect better from computer-related publications (especially technical ones). I have always been proud to call myself a hacker, and I find it offensive to be grouped in with a bunch of criminals. I think *PC Tech Journal* should establish

a policy that the word be used properly. Hackers make up a large part of your readership, and it would be best not to offend them.

John McNamee
Canoga Park, CA

An informal survey taken at my law firm indicated that the term hacker *refers to someone operating in the penumbra of what is legal. A similar survey taken at* PC Tech Journal *indicated that the term connotes nothing less socially acceptable than perhaps excess enthusiasm. Webster's is no help at all, defining hacker as someone who is inexperienced. I suppose that is one reason English is not a good computer programming language.*

I sympathize with Mr. McNamee, but it appears that hacker is acquiring, at least in general usage, a connotation of marginally lawful activity.

—Max Stul Oppenheimer

PASCAL WORKAROUND

Regarding the Microsoft Pascal bugs Ted Forgeron alerted us to ("Pascal Bugs," May 1985, p. 199), there is a workaround for the uninstalled AT 80287 that requires neither the IBM libraries nor relinking.

The README file of 3.2 Pascal contains instruction on suppression of the 8087 that works just fine for suppression of the 80287 as well. The DOS SET command is used to place the phrase "N087= " in the environment prior to execution. Note that at least one blank must follow the equals sign, and programs must have been linked with MATH.LIB, which automatically emulates the math coprocessor if its presence is not detected.

Chet Floyd
Manhattan Beach, CA

Mr. Floyd correctly points out that the SET command can be used to force floating emulation with version 3.2 of Microsoft Pascal. However, there are two situations in which use of SET is not necessary. First, if you link with ALTMATH.LIB or DECMATH.LIB, you need not worry about an uninstalled 80287, because neither of these libraries uses or looks for the coprocessor. Second, versions of Microsoft Pascal before 3.2, such as 3.13, do not recognize the N087 environment variable and cannot be fixed with Mr. Floyd's workaround.

The good news is that Microsoft has fixed this bug in its version 3.3.

—Ted Forgeron

ASSEMBLY TROUBLES

I was quite interested to read the letter and response about the "DOS Redirection Bug" in your May 1985 issue (Letters, p. 20). I have reported on this bug and have made the additional observation that the user can display the contents of a file with an embedded EOF by using the COPY command with the /B option.

The embedded EOF can be edited out with EDLIN if EDLIN is called with the /B option and the Ctrl-V convention is used for the editing, but EDLIN will insist on always placing an EOF at the end of the file when it is saved. One way to get rid of the closing EOF is to use a pornographic application of the COPY command, such as

```
COPY TESTFILE+,/B
```

You cannot worry too much about the error message produced. Be sure to put the /B after the plus and comma.

I also have a question about another topic. Am I the only person who has had a difficult time assembling the program PACKDIR.ASM ("Dipping into Directories," Ted Mirecki, February 1985, p. 67)? Both versions of the IBM Macro Assembler choked on the lines in PROCEDURE OPEN of the form CMP BYTE PTR FCB2+1.ATTR,10H. "Syntax error" was reported in each case. I found, however, that if I changed the lines to the form CMP BYTE PTR FCB2.ATTR+1,10H that everything seemed to proceed smoothly. Being somewhat of a novice with assembly language, I would appreciate some confirmation that these statements are equivalent before I fool around too much and end up destroying something on my disk.

Bob Stephan
Pebble Beach, CA

I am not sure which two versions of the IBM Macro Assembler Mr. Stephan is referring to, but both my and PC Tech Journal's *copies of version 1.0 had no trouble with these statements. IBM's version 2.0 and Microsoft's version 3.0, however, are a different story; they do in fact complain about this construction. Mr. Stephan's solution is the correct one, as both ways of expressing this kind of operand are logically equivalent; whether to accept one, the other, or both is an arbitrary decision on the part of the assembler designer.*

This is only one of several differences between the newer and older versions of the IBM/Microsoft Assembler. Another is the way that segments are ordered in the load module: the old ver-

sion always arranged them alphabetically; the new ones offer the programmer a choice, with the default being the order of occurrence in the source. Therefore, if PACKDIR.ASM is assembled with either IBM 2.0 or Microsoft 3.0, the /A parameter must be specified at assembly time to force alphabetic ordering. Without it, the program assembles and links, but cannot be converted to a .COM file with EXE2BIN.

A review of the new assemblers, with descriptions of differences from the older version, will soon be published by PC Tech Journal.

—Ted Mirecki

SNOBOL4+ INQUIRIES

From the number of telephone calls we have received, I know the SNOBOL4 article in the January 1985 issue of *PC Tech Journal* ("SNOBOL4," Richard Larson, p. 32) sparked considerable interest among your readers. Due to a mix-up, we were not sent labels for inquiries directed to the SNOBOL4+ reader service number listed at the end of the article. Could you mention to your readers that we are not ignoring them, but rather that we did not know who they were?

SNOBOL4+ is now being published by Prentice-Hall, Inc. The software is accompanied by a 248-page book, divided equally between tutorial and reference manual. The minor I/O differences mentioned by Richard Larson have been corrected. SNOBOL4+'s price is still $95; it can be ordered from: Prentice-Hall, Inc., Business and Professional Books Division, 200 Old Tappan Road, Old Tappan, NJ 07675. The telephone number is 800/624-0023.

Mark B. Emmer
President
Catspaw, Inc.
Salida, CO

We apologize for the difficulty you had with the reader service number. That number is meant to be just what it says: a reader service. We regret that it did not live up to its name in this case.

—WF

ELEGANT PACKING

While I was writing an application that involved performing some date comparisons and storing of the date in a data record, I turned to Patrick Finan's article "PACKing the Date and Time" (January 1985, p. 46). At first it seemed an ideal procedure until I gave it some more thought. I wanted to know, not only which of two dates occurred earlier, but also how many days apart were the two dates. I wrote my application in C so I used Kernighan and Ritchie's algorithm for determining the day of the year that is listed on page 104 of their book, *The C Programming Language*. I then ORed in the year base 1980 in the 7 high bits. The result yielded me a 16-bit date, with plenty of room for the year and a more elegant method of comparing dates.

Karen B. Thibodeau
Dallas, TX

ERRATA

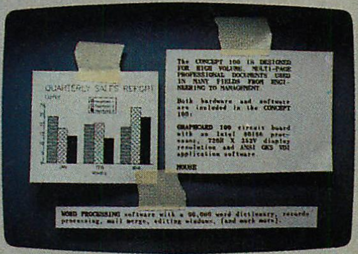
W. David Schwaderer's article, "CRC Calculation" (PC Tech Journal, *April 1985, p. 115), should have carried a reference to an earlier article on the same subject, "Byte-wise CRC Calculations"* (IEEE Micro, *June 1983, p. 40) by Aram Perez. We regret the omission.*

Photographer Paul D'Innocenzo should be credited for his work in "Xenix for the XT" (Augie Hansen, PC Tech Journal, *June 1985, p. 129).*

TEXT PLUS GRAPHICS, MINUS SCISSORS AND PASTE: THE $2195 PC SOLUTION!

Now you can produce full-function text-and-graphics for reports, manuals, color overheads and slides right on your PC XT/AT. All you'll cut out is the scissors and paste! The CONCEPT 100 System lets you create text on a Wang-like word processor, complete with 90,000-word dictionary and list processing. You can easily create charts and drawings. Or import text or graphics from other systems. Then, use electronic cut-and-paste to combine them on-screen just as you'll see them in hard copy.

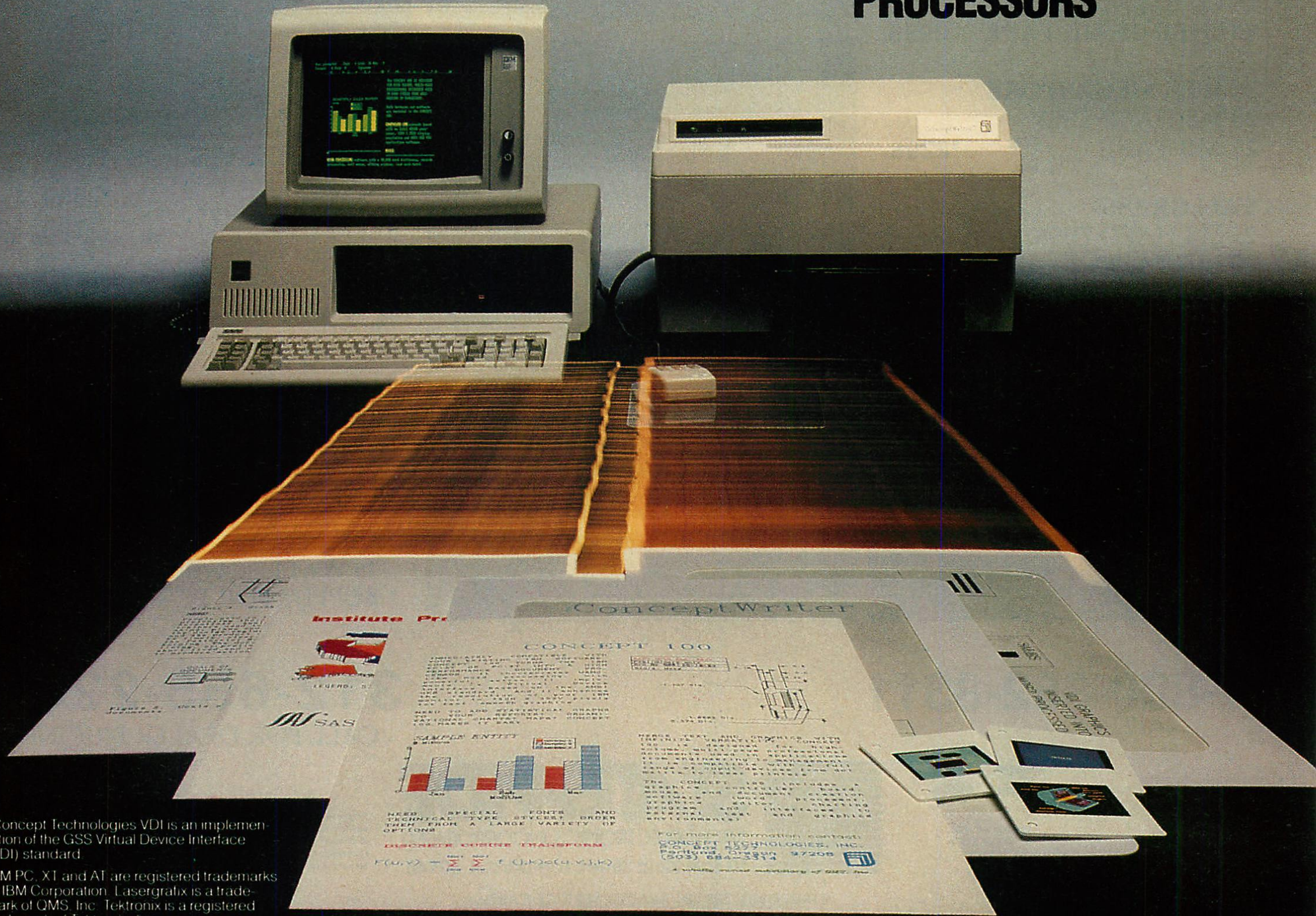
CONCEPT 100™ is based on the ANSI GKS* standard using the Virtual Device Interface. It offers Tektronix terminal emulation and output options like the ConceptWriter™ or Lasergrafix laser printers. In short, it's a personal publishing system at an unprecedented low price. Now available from your local distributor.



Call 1-800-631-2692 for more information.

CONCEPT TECHNOLOGIES, INC.
A QMS Company
P.O. Box 5277, Portland, OR 97208

THE DOCUMENT PROCESSORS©



*Concept Technologies VDI is an implementation of the GSS Virtual Device Interface (VDI) standard.

IBM PC, XT and AT are registered trademarks of IBM Corporation. Lasergrafix is a trademark of QMS, Inc. Tektronix is a registered trademark of Tektronix, Inc.

CIRCLE NO. 117 ON READER SERVICE CARD

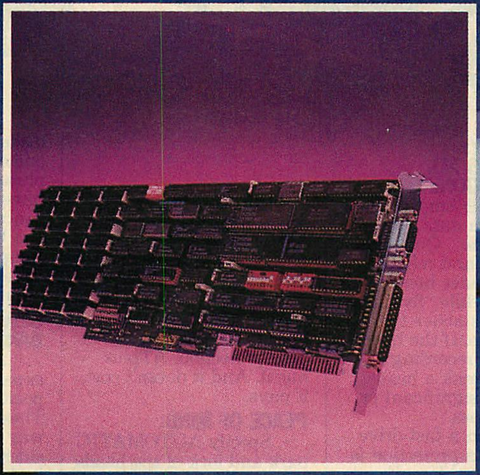
Alloy makes your PC multi-user.

Introducing Alloy's PC-SLAVE/16. The multi-user PC strategy that doesn't require additional PCs.

PC-SLAVE/16 transforms your IBM PC into a multi-user, multi-processor system with shared data access and communication between users. All you need is a dumb terminal and a PC-SLAVE/16 board for each user.

And with Alloy's PC-X BUS, you can add up to 31 users. So whether you're adding your secretary or 31 colleagues, Alloy's PC-SLAVE/16 can do it at a much lower cost and with higher performance.

For any system requiring more than one PC, PC-SLAVE/16 is more powerful and less expensive. Each board comes complete with its own 8 MHz, 8088 micro processor and dedicated memory. PC-SLAVE/16 eliminates the need to add costly PCs. And it is ten times faster than LANs (Local Area Networks) at half the cost. Best of all, it's available right now. See PC-SLAVE/16 in action at your local dealer. Or call Alloy today at (617) 875-6100.



ALLOY

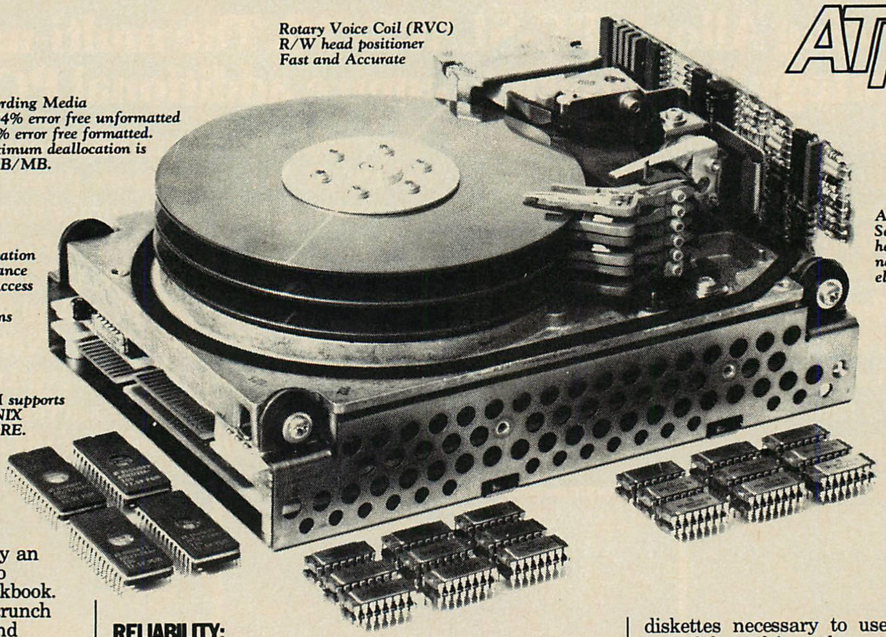
ALLOY
Computer Products, Inc.

Alloy Computer Products, Inc., 100 Pennsylvania Avenue, Framingham, Massachusetts 01701. (617) 875-6100, TWX: 710-346-0394
In Europe: Alloy Computer Products (Europe) Ltd., Cirencester, Gloucestershire, England. Tel: 0285-68709, Tlx: 43340

CIRCLE NO. 112 ON READER SERVICE CARD

BEFORE YOU GET ON THE WRONG TRACK, READ THIS ABOUT THE ONLY ROM-BOOTABLE, PLUG-COMPATIBLE, XENIX SUPPORTING, HEAD-CRASH PROTECTED, HIGH-PERFORMANCE HARD DISK UPGRADE KIT FOR YOUR IBM PC-AT.

ATplus T.M.



THE BALANCING ACT OF 1985.

You didn't buy an IBM PC-AT just to balance your checkbook. You bought it to crunch lots of numbers and words, in the shortest possible time. A labor saver. A time saver. Hence, a money saver.

So, do your part for effective money management; hard disk storage is no place to be penny-wise and pound foolish.

TELL 'EM YOU NEED HIGH SPEED AND DATA PROTECTION.

These and other important features do add cost, but that makes a premium drive.

Anything that can be made, can be made cheaper, sell for less, offer lower performance, and probably die young.

Remember, usually you get what you pay for, and you ALWAYS get what you don't.

ALL HARD DISKS ARE NOT CREATED EQUAL.

There are vast differences in the speed and reliability of Winchester hard disks. Since the IBM PC-AT is an incredibly fast machine, a slow drive can make an AT run like an XT.

So, before you get stuck with a slow drive in your AT, save your boss two grand and buy an XT.

Or better yet, buy the AT and avoid any drive with Access Times over 40 milliseconds.

RELIABILITY: WHERE HAS ALL THE DATA GONE?

Now tell 'em the drive must have a data protection scheme. One that's easy to use and reliable.

Winchester heads read and write while "flying" a few microns above the data surface. If the heads contact the recording media, you risk a head crash, and significant or total data loss.

So, even a fast drive without data protection is virtually worthless. Frankly, we'd rather sleep at night.

BEWARE OF USER-DEPENDENT PROTECTION SCHEMES.

Some drives have a safe landing zone for the heads, but you need to call a separate program to send 'em there. If you don't call that program, and most folks won't, the heads in these drives ALWAYS land on data when powered down.

The slightest bump or vibration can move the heads, wiping out those data tracks. And the R/W heads can become contaminated, thus increasing the error rate, slowing down average access until the whole drive fails.

Consequently, those drives offer a very high risk of head crashes, a false sense of security, and little else.

What's your data worth? $200? $400? Specify AUTOMATIC data protection. ATplus has it. And it doesn't cost, it pays.

PEACE OF MIND.

Specify AUTOMATIC park and lock of the heads on power down.

This system provides unparalleled head crash protection, by sensing power loss to the drive, and retracting the heads to a dedicated landing zone before they can land on your data.

Since this is 100% automatic, user-dependent risk is eliminated.

OUR DRIVES HAVE ALL BEEN TO BOOT CAMP.

Avoid drives that CLAIM PC-AT compatibility but can't BOOT the AT. By the time you juggle the diskettes necessary to use one of those drives, the phrase "user-hostile" will have deep personal significance.

We believe that computers ought to serve people, not the other way around.

BEWARE OF THE BARGAIN BAND-SCHLEPPER.

Avoid drives with inexpensive Band-Stepper positioner technology. These were pretty good way back in 1980, considering that's all anyone had. But by today's standards, they're inaccurate and very mechanical.

They waste time looking for the right track to read or write. And they're worth no where near the price you'll pay for 'em –in more ways than one.

THE HIGH TECH SOLUTION.

Specify state-of-the-art ROTARY VOICE COIL (RVC) head positioner and CLOSED LOOP SERVO technologies.

This system uses a DEDICATED SERVO SURFACE (DSS) for continuous head location information.

The RVC and DSS use a large scale microprocessor to translate new track-seek commands into current that is applied directly to the RVC.

This moves and holds the R/W heads at the exact track intended.

Speed is dramatically increased since head movement is created electronically, with minimal mechanical interfacing.

BESIDES BEING THE SOLE SOURCE.

Specify names you know and can trust. Like Control Data Corporation, a world leader in hard disk technology. And CORE INTERNATIONAL, a growth company with over ten years in computer technology, design and manufacturing.

ATPLUS: IDEAL FOR THE SPECS MANIAC.

ATplus High Performance Hard Disk Upgrade kits, available in 20, 30, 40, 56, and 72 MB capacities.

They have ALL the above features and none of the shortcomings of "bargain" drives.

And your AT will accept any two ATplus drives, for up to 144MB of bootable, reliable, DOS and XENIX supporting high-performance on-line storage, all inside the box.

TELL YOUR DEALER...

That only ATplus Hard Disk Upgrade Kits from CORE will do.

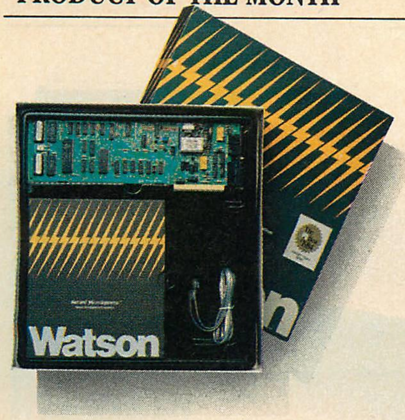
If he or she doesn't have 'em, we'll gladly set them up as ATplus authorized dealers.

CORE INTERNATIONAL

542 S.E. 5th Avenue • Delray Beach, Florida • 305/276-3929

ATplus is a trademark of CORE INTERNATIONAL, Inc.

CIRCLE NO. 166 ON READER SERVICE CARD



The Watson System

It does not profess to be more than it is; no claims of speech recognition—just a strong voice/data management system.

Good morning. It is 9:07 a.m. You have reached the Zircon Corporation Field Service Debriefing Center. Please enter your employee number."

The caller presses seven Touch-Tone buttons.

"Thank you. You may press 1 to debrief your current call, 2 to receive your next call, 3 to leave a message, 4 to check your mailbox, 5 to speak to a dispatch specialist, or pound sign to hear this menu again."

Welcome to the age of digital speech management. The entire imaginary application introduced above can reside on an ordinary PC/XT or AT with 20MB or more of fixed-disk space. The technology exists today for less than $1,000. It's called the Watson Voice/Data Management System, and *PC Tech Journal* would like to present it as August's Product of the Month.

Watson is a hardware-software product consisting of a full-length PC circuit board and driver software. The board contains digital-to-analog and analog-to-digital hardware, a Cermetek 300/1200 baud modem, and a TI TMS320 digital signal processor CPU to control it all. Two modular telephone jacks allow the telephone to perform both local and remote speech I/O for the board. Included in the package price of $698 are the board, drivers, and PC-Talk configured for the Watson modem, which is Hayes-compatible.

The main purpose of the Watson driver software (some of which is on ROM on the board and some of which is disk-based) is to store speech as data in a disk file and later to re-create speech from the data in those disk files. The fidelity of reproduced speech is strikingly good—better than most inexpensive tape-based telephone answering machines. The end-user application sold with Watson is, in fact, a totally digital answering machine with a twist: callers can type in a code number and retrieve messages left for them. The system becomes a simple voice-oriented computer bulletin board system.

The high-reproduction fidelity has a cost: a minute of digitized speech occupies about 240KB. Higher data compression can be achieved (down to 60KB per minute) with lower fidelity and a higher noise level. These storage requirements might have seemed prohibitive even a year ago, but today, with 20MB disk subsystems selling for less than $700, serious interactive voice/data applications can be built on PC systems costing less than $4,000.

PRODUCT NAME
The Watson Voice/Data Management System

COMPANY
Natural Microsystems Corporation

ADDRESS
6 Mercer Road,
Natick, MA 01760

TELEPHONE
617/655-0700

PRICE
Watson: $698
VARI: $198
VIS: $298

Two development environments are available for the Watson system. At the lowest level is the Value Added Reseller's Interface (VARI), which offers complete control of the Watson digital signal processing hardware through a library of Lattice C functions. VARI is interrupt-driven and permits monitoring of telephone and modem lines in the background while executing another Watson application in the foreground. VARI is available to developers for a one-time charge of $198.

Although VARI offers the most versatility through complete control of the hardware, most developers have chosen the other environment, the Voice Information System (VIS). VIS is preferable because it focuses on interactive voice/data systems and allows an application to be constructed in much less time than VARI. VIS is a command interpreter that lets a developer build "card files" of commands and stored voice messages. The card file, when executed, becomes the application.

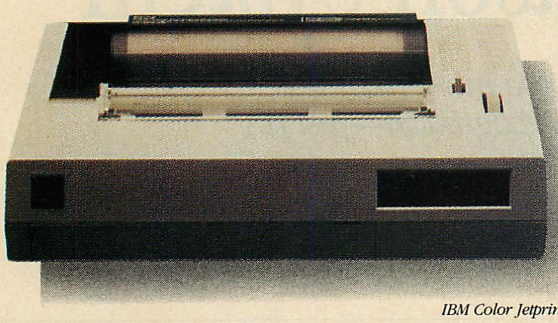
VIS supports an accumulator and a number of registers for storing and manipulating whole numbers and characters including date and time information. It includes commands to play messages, dial the telephone, sense phone line states (dial tone, busy, voice, etc.), listen for Touch-Tone signals, wait for specified time intervals, branch to another command based on user input or contents of registers, and quite a few others. VIS cannot access Watson's modem (VARI must handle that) and VIS applications can run only in the foreground. Within the realm of interactive telephone voice applications, however, VIS can handle almost any machine-user scenario that does not involve actual speech recognition.

Watson is a significant product in one sense because it does not try to be more than it is. In the perception of the Natural Microsystems Corporation, speech recognition, like artificial intelligence, is a concept existing more in the public imagination than in today's inexpensive computer systems.

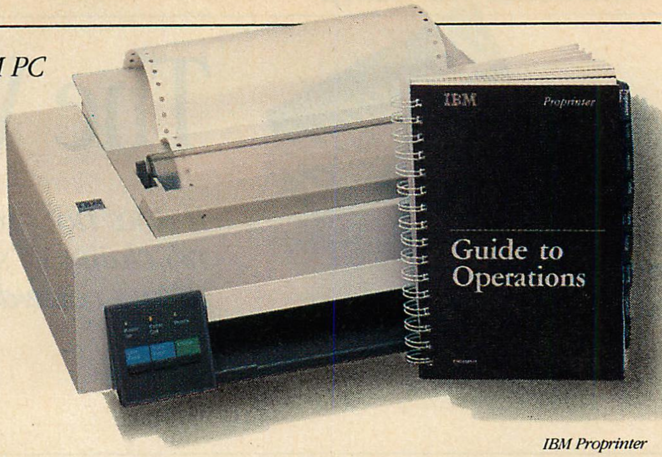
For less than twice the list price of a Hayes 1200 modem, Watson offers Hayes compatibility, speech digitization, and complete telephone control. The software is well-documented and optimized to save the developer's time.

Watson will intrigue anyone who is interested in interactive voice/data applications. (Callers can interact with a Watson system by dialing 800/6-WATSON.) It was among the first, and we feel it to be the best voice/data development tool in its price class.

Hardware, software, and other developments for the IBM PC



IBM Color Jetprinter



IBM Proprinter

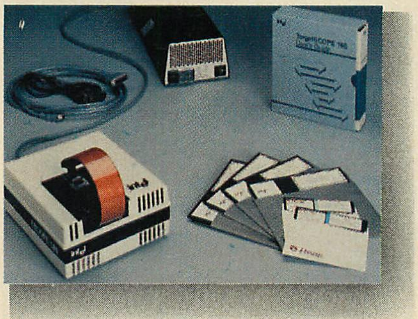
HARDWARE

From **IBM** come two new printers: the **IBM Proprinter** and the **IBM Color Jetprinter**. The Proprinter, a dot-matrix printer, has draft, correspondence, and near-letter-quality modes. In draft mode, it prints at a speed of 200 characters per second. Continuous-form paper, fed from the top, can be left in place while single-sheet paper or envelopes are inserted in front.

The Color Jetprinter can print high-quality graphics or near-letter-quality text in as many as seven colors on transparencies and a variety of paper. It provides an eight-inch writing line, pin-feed paper mechanism, draft and near-letter-quality modes, superscript and subscript support, and full foreground, background, and surface color support. Proprinter, $549; Color Jetprinter, $745.
IBM Corporation, Information Systems Group, 900 King Street, Rye Brook, NY 10573; 914/934-4822
CIRCLE 306 ON READER SERVICE CARD

Also from **IBM** come two new industrial computers. The **IBM 7531 Industrial Computer** operates as a floor-standing unit, and the **IBM 7532 Industrial Computer** is designed to mount in an industry-standard, 19-inch rack. The systems are functionally identical and offer extended protection against temperature extremes, vibration and shock, voltage transients, and particulates. Both computers include 512KB of standard memory (expandable to 3MB), a 1.2MB diskette drive, seven expansion slots, clock/calendar, and ROM-based BASIC. Both systems require a video display adapter and display for video output. 7531 Industrial Computer, $6,145; 7532 Industrial Computer, $6,370.
IBM Corporation, Manufacturing Systems Products, P.O. Box 1328, Boca Raton, FL 33432; 305/982-2659
CIRCLE 307 ON READER SERVICE CARD

Intel has announced **TargetSCOPE 186**, a high-level, language-debugging tool for the PC/XT and PC/AT that lets users symbolically debug programs in PL/M-86, Pascal-86, FORTRAN-86, C-86, and assembly language-86 for the iAPX-186 family of microprocessors. TargetSCOPE 186 allows software engineers to download, observe, and modify high-level programs running on 80186- and 80188-based target systems. Software engineers can use TargetSCOPE to debug in-target I/O-related software or other low-level subroutines. $5,495 for package, including system and software.
Intel Corporation, 5200 N.E. Elam Young Parkway, Hillsboro, OR 97123; 503/681-2010
CIRCLE 313 ON READER SERVICE CARD



TargetSCOPE 186

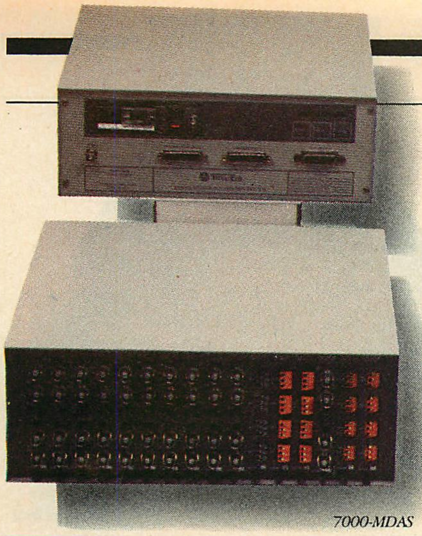
Okidata has announced the **Microline 192** and its wide-column companion, the **Microline 193**. Both dot-matrix printers are available in specially configured, IBM-compatible models; these models come with free software that allows the PC/XT user to produce type styles similar to courier, italic, gothic, and scientific characters, as well as to design and download custom typefaces. These printers offer bidirectional printing, 8KB buffer, high-resolution mode, and proportional spacing. Microline 192, $499; Microline 193, $699.
Okidata, 532 Fellowship Road, Mt. Laurel, NJ 08054; 609/235-2600
CIRCLE 314 ON READER SERVICE CARD

A new product for the high-performance CAD and professional graphic arts industries has been announced by **Orchid Technology**. The **Turbo Graphics Controller (TGC)**, which is functionally compatible with the IBM Professional Graphics Controller, runs 4 to 25 times faster than IBM's product. The TGC offers 640-by-480 pixel display resolution and can run on a less expensive display than IBM's. It writes directly to screen memory (bit-map), and has RAM designed specially for video applications. Approximately $2,000.
Orchid Technology, 47790 Westinghouse Drive, Fremont, CA 94539; 415/490-8586
CIRCLE 315 ON READER SERVICE CARD

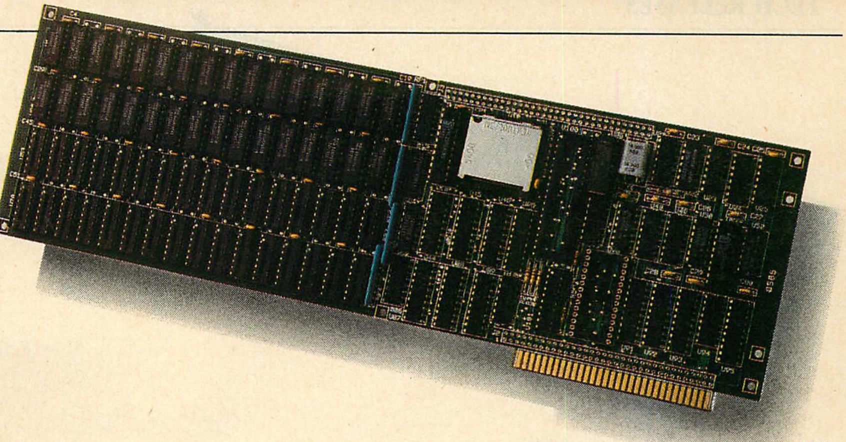
Symphony Link, an add-in product that enables users of Lotus Symphony to communicate in a synchronous environment, has been announced by **Digital Communications Associates, Inc.** and **Lotus Development Corporation**. Symphony Link operates on a PC/XT or PC/AT that is also equipped with Digital Communications' IRMA, IRMAkey/3270, or IRMAline/IRMAlette. The product lets Symphony users communicate with a mainframe in an IBM 3270 environment by allowing the PC to emulate an IBM 3278/79 terminal. $395.
Digital Communications Associates, Inc., 303 Technology Park, Norcross, GA 30092; 404/448-1400
CIRCLE 316 ON READER SERVICE CARD

Lotus Development Corporation, 55 Cambridge Parkway, Cambridge, MA 02142; 617/577-0500
CIRCLE 317 ON READER SERVICE CARD

GTCO Corporation has introduced the **FP1-1622 Photoplotter** for use with microcomputer-based PCB CAD systems. The FP1 Photoplotter creates one-to-one precision artwork for printed circuit boards with line width and spacing as fine as 12 mils. The plot area is a flat



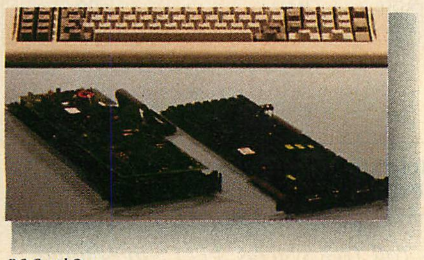
7000-MDAS



Pfaster286

bed with vacuum hold-down; it accommodates plots of up to 15.5 by 21.5 inches. Price: $25,000.
GTCO Corporation, 1055 First Street, Rockville, MD 20850; 301/279-9550
CIRCLE 319 ON READER SERVICE CARD

New Media Graphics Corporation's **PC-GraphOver** turns a PC into an interactive video system that controls videodisk playback and superimposes high-resolution (640-by-400) color graphics on video backgrounds. It is the only graphics overlay generator for video that runs standard PC software, such as Lotus 1-2-3, without modification. It offers 16 full colors that can expand to 4,096 simultaneous colors. $1,990.
New Media Graphics Corporation, 279 Cambridge Street #5, Burlington, MA 01803; 617/272-8844
CIRCLE 318 ON READER SERVICE CARD



PC-GraphOver

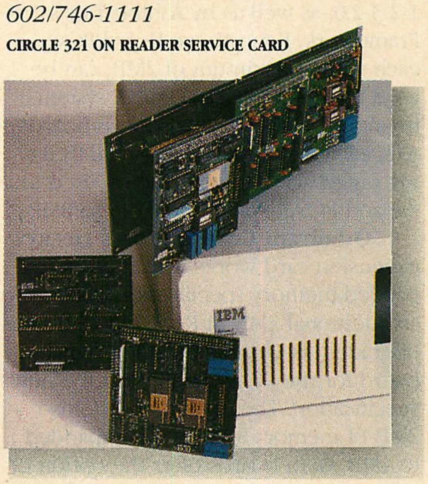
A new modular data acquisition system from **Transera Corporation**, the **7000-MDAS**, has its own operating system and can function independently of the host, capture large arrays of analog and digital data at high speeds, and coordinate complex acquisition and control routines. The realtime operating system supports multitasking and provides high-level communication with any attached host. $2,000 for four channels (up to 64 channels available).
TransEra Corporation, 3707 North Canyon Road, Provo, UT 84604; 801/224-6550
CIRCLE 320 ON READER SERVICE CARD

Pfaster286, from **Phoenix Computer Products Corporation**, is the first 80286-based, add-on board to enable the PC and PC/XT to process data faster than a PC/AT without losing the functionality of the native 8088. By running the 80286 at an 8-mHz clock rate with no wait states (compared to the AT's 6-mHz with one wait state), this board speeds up the data processing power of the PC or XT to three times that of an XT or up to 60 percent faster than an AT. $2,395 (1MB RAM).
Phoenix Computer Products Corporation, 1416 Providence Highway, Suite 220, Norwood, MA 02062; 617/762-5030
CIRCLE 312 ON READER SERVICE CARD

Sysgen, Inc. has announced the **SI536**, a multifunction SCSI disk/tape controller that offers both QIC-36 tape and ST506 Winchester disk interfaces on one board. The SI536 controls up to two ST506/412 hard-disk drives and one QIC-36 tape drive for hard-disk back-up. Up to 256KB of optional cache memory is available, boosting performance 30 to 40 percent. $400 in OEM quantities.
Sysgen, Inc., 47853 Warm Springs Blvd., Fremont, CA 94539; 415/490-6770
CIRCLE 322 ON READER SERVICE CARD

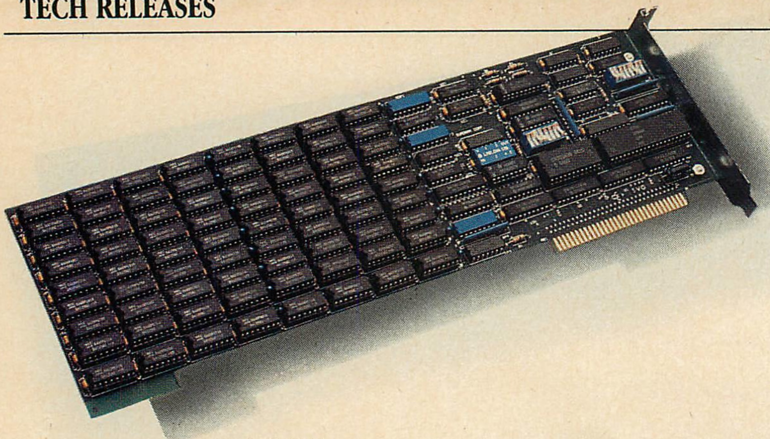
A new speech input system called **Pronounce** has been introduced by **Microphonics Technology Corporation**. Pronounce offers the user an unlimited number of vocabulary files of 128 words or short phrases. Each word or phrase may be associated with up to 255 keystrokes, thus forming a macro command capability. Pronounce consists of a board that fits into any PC/XT or PC/AT, software with predefined vocabularies for WordStar and Lotus 1-2-3, and a microphone. $895.
Microphonics Technology Corporation, 234 S.W. 43rd Street, P.O. Box 7458, Renton, WA 98057; 206/251-9009
CIRCLE 323 ON READER SERVICE CARD

The **PCI-20000** from **Burr-Brown Corporation** is a system that provides plug-in component modularity, allowing users to specify an exact instrumentation while still expanding or changing that instrumentation at any time. The PCI-20000 consists of a bus-compatible carrier board, which plugs into an expansion slot of the PC/XT and PC/AT, and a family of instrument modules, which plug into the carrier. The carrier provides computer interface, power, and intermodule communications. Each carrier accommodates three instrument modules. Carrier board, $295; instrumentation modules, from $199.
Burr-Brown Corporation, P.O. Box 11400, Tucson, AZ 85734; 602/746-1111
CIRCLE 321 ON READER SERVICE CARD



PCI-20000

The **TIGER-32** add-on board from **DFE Electronic Data Systems** allows PC/XT and PC/AT users to tap into the power of a DEC VAX, with a 10-mHz coprocessor running XENIX 3.0 applications for as many as 16 users on a full 32-bit data bus. The board uses one slot and comes with 512KB of RAM, 6-mHz clock, and a two-user XENIX 3.0 system. $2,095.
DFE Electronic Data Systems, 1250 Oakmead Parkway, Suite 210, Box 3599, Sunnyvale, CA 94086; 408/730-2652
CIRCLE 324 ON READER SERVICE CARD



Quadram's LIBERTY

Kaypro 286i

A number of new products have been announced to support the new Lotus/Intel expanded-memory specification, which allows software to address RAM in excess of the standard 640KB (up to a maximum of 8MB). See the table at right for memory capacity and pricing.

The **RAMpage!** board from **AST Research, Inc.** enhances the specification to provide four times as much total area above 640KB. The enhancements, which were developed with Ashton—late and Quadram Corporation provide for a more flexible paging scheme.

The **Above Board/PC** and **Above Board/AT** are two new boards from **Intel Corporation**. These new boards are supported by Lotus' Symphony 1.1 and 1-2-3 2.0, as well as by Ashton-Tate's Framework. Four Above Board/PCs, each with a maximum of 2MB, can be used to expand PC and PC/XT system memory up to 8MB; two Above Board/ATs, each with a maximum of 4MB using piggyback memory, can be used to expand PC/AT system memory to 8MB.

Quadram Corporation's LIBERTY expansion card works with the new expanded-memory specification to provide an additional 2MB of RAM. Quadram has upgraded its **QuadMEG-AT** expansion board for the PC/AT to support the new specification. The main board holds 2MB of memory and can be expanded to 4MB. Both boards offer the option of using split memory mapping to use a portion of the RAM on the boards to complete the system memory of 640KB.

STB Systems, Inc. has announced the **Memory Companion/PC**, an expansion board that provides the PC/XT with 2MB of additional memory. Four Memory Companion/PCs may be installed in a single system, increasing RAM to 8MB.

The **JRAM-3**, from **Tall Tree Systems**, allows users to access up to 2MB of RAM on a single board; four boards may be used in one system to expand the total RAM to 8MB.

TABLE: *Memory-expansion Boards*

COMPANY and PRODUCT	BASE MEMORY	BASE PRICE	MAXIMUM MEMORY/ BOARD	PRICE
AST Research RAMpage!	256KB[a]	$595	n/a	
Intel PCEO Above Board/PC	256KB[a]	495	2MB	$1,395
Intel PCEO Above Board/AT	512KB[a]	795	2MB 4MB	1,495 2,690
Quadram Corp. LIBERTY	64KB[b] 512KB[b] 1MB[b]	395 595 895	2MB	1,295
Quadram Corp. QuadMEG-AT	128KB[c] 512KB[c]	445 595	2MB 4MB	1,295 2,685
STB Systems Memory Companion/PC	256KB[a]	395	2MB	2,145
Tall Tree Systems JRAM-3	256KB[a]	399	2MB	699

[a]*Configured with 256K chips*
[b]*Configured with 64K chips*
[c]*Configured with 128K chips*

These products support the new Lotus/Intel expanded memory specification, which allows software to address RAM in excess of the standard 640KB.

AST Research, Inc., 2121 Alton Avenue, Irvine, CA 92714; 714/863-1333

CIRCLE 368 ON READER SERVICE CARD

Intel Corporation, 5200 N.E. Elam Young Parkway, Mail Stop TOC-03, Hillsboro, OR 97123; 503/629-7369

CIRCLE 311 ON READER SERVICE CARD

Quadram Corporation, 4355 International Blvd., Norcross, GA 30093; 404/923-6666

CIRCLE 310 ON READER SERVICE CARD

STB Systems, Inc., 601 N. Glenville, Suite 125, Richardson, TX 75081; 214/234-8750

CIRCLE 308 ON READER SERVICE CARD

Tall Tree Systems, 1121 San Antonio Road, Palo Alto, CA 94303; 415/964-1980

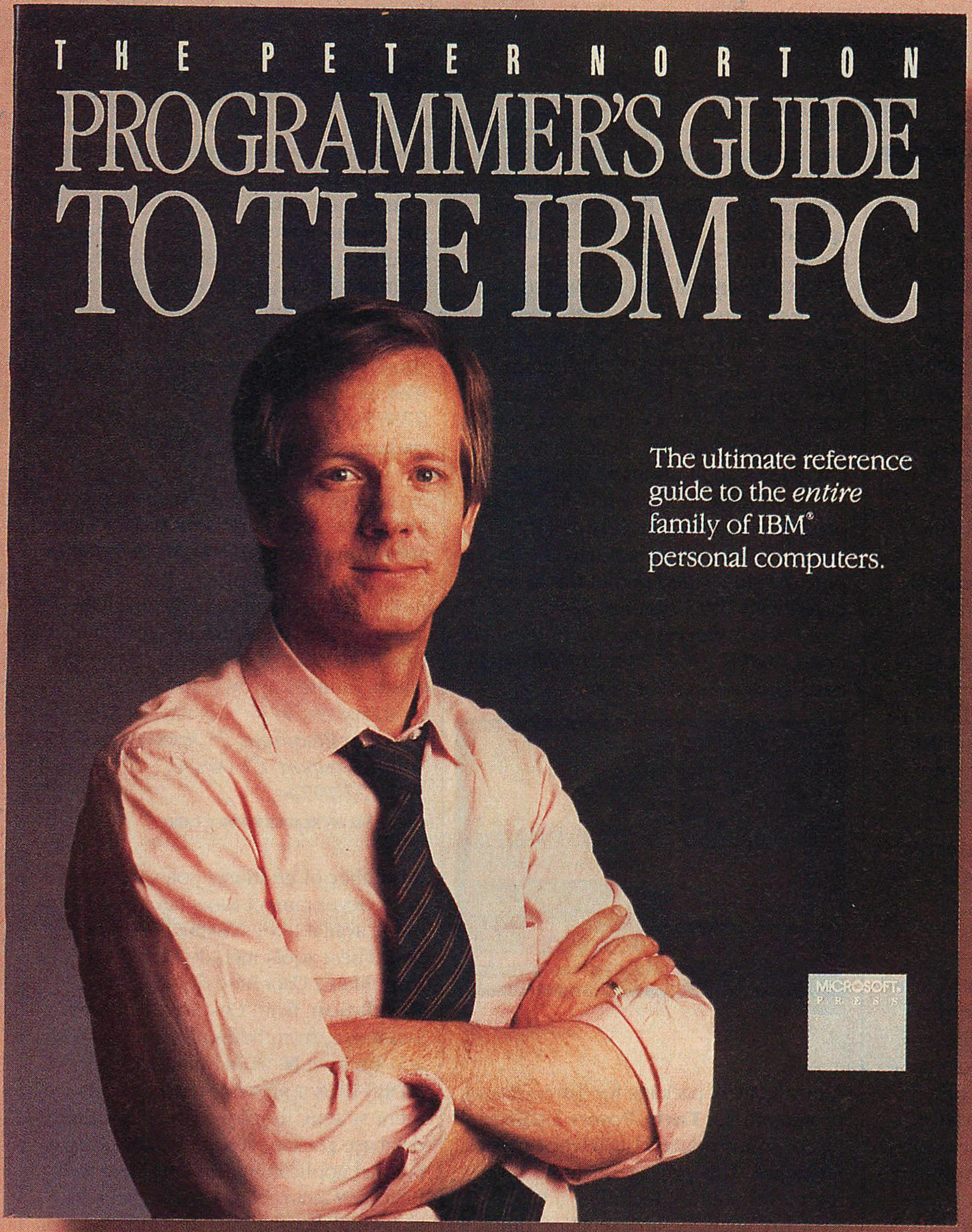
CIRCLE 309 ON READER SERVICE CARD

Kaypro Corporation has announced the first IBM PC/AT-compatible computer. The **286i** has an 80286 microprocessor with a clock speed of 6.0 mHz; 512KB RAM standard (expandable to 640KB on the main board and to 15MB using additional boards); two high-density floppy-disk drives; eight I/O expansion slots; a standard AT-compatible color graphics card with RGB output; and a selection of software. $4,550. *Kaypro Corporation, P.O. Box N, Del Mar, CA 92014; 619/481-4300*

CIRCLE 325 ON READER SERVICE CARD

MICROSOFT PRESS ■ PETER NORTON ■ THE IBM PC

AN UNBEATABLE COMBINATION



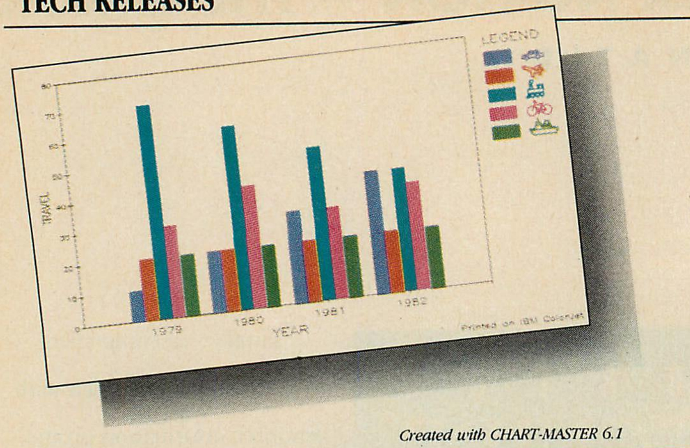
Finally... a comprehensive reference guide rich with resource information on the *entire* IBM family of personal computers: the PC, AT, XT, PCjr, and Portable PC.

The Peter Norton Programmer's Guide to the IBM PC is your ultimate guide to programming for the IBM PC environment. It's a gold mine of insights, techniques, technical data, and quick reference charts, all emphasizing the architectural similarities and differences among IBM's five personal computers. Whether you're an experienced or a novice programmer working in assembly language, Pascal, BASIC, or C, you'll find invaluable information for developing portable, effective professional applications.

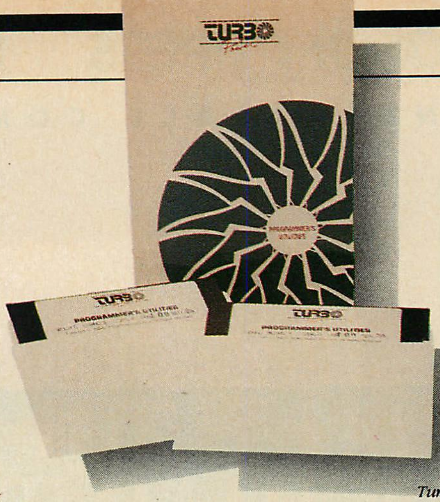
By Peter Norton, recognized authority on IBM technology, creator of the acclaimed *Norton Utilities*, and noted architect of the now-famous philosophy of simple, clean programming.

$19.95 Available now wherever books and software are sold.

Microsoft Press 10700 Northup Way Bellevue, Washington 98004



Created with CHART-MASTER 6.1



TurboPower Utilities

SOFTWARE

Two products designed to make the features of the Hewlett-Packard Laserjet printer easily accessible to users and applications programs have been announced by **Network Technology Corporation**. **LASERTEX** is a set of memory-resident utilities that allows users to change page formats, type styles, type fonts, type sizes, and number of copies; in addition, they can integrate text and graphics on the same page. LASERTEX is compatible with WordStar and other word processors.

The **GUTENBERG Laser Composition System** includes all LASERTEX utility functions plus electronic cut-and-paste functions, custom fonts, Epson printer emulation, templates for standard formats, and dot, tone, line, and texture patterns. GUTENBERG allows fully collated document sets to be produced, complete with printed covers. LASERTEX, $129.95; GUTENBERG, $695.00.
Network Technology Corporation, 6825 Lamp Post Lane, Alexandria, VA 22306-1321; 703/765-4506

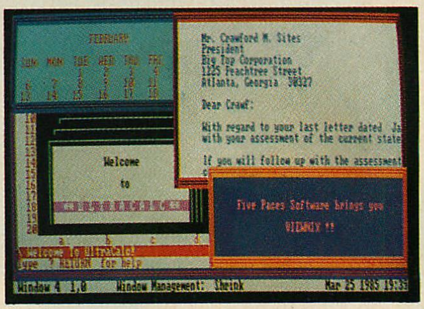
CIRCLE 335 ON READER SERVICE CARD

Case Technology, Inc. has introduced the **CT2000 CAE Design System**, a CAE software package for designers of integrated circuits and printed circuit boards. The CT2000 includes a fully implemented version of the advanced structured computer-aided logic design (SCALD) tools. The system also provides a structured graphics editor, which permits the creation of user-defined symbols and components and the use of other such components that have been stored in the design database. The CT2000 interfaces easily with other CAE/CAD tools. $5,200.
Case Technology, Inc., 633 Menlo Avenue, Menlo Park, CA 94025; 415/322-4057

CIRCLE 336 ON READER SERVICE CARD

Viewnix, a windowing interface for UNIX-based microcomputers, is now available from **Five Paces Software, Inc.** Viewnix is currently being shipped for the PC/AT and PC/XT running XENIX. Viewnix allows the user to configure as many as 10 windows on the screen of an AT, each of which can contain any standard XENIX application. The windows can be expanded, contracted, or moved about the screen to suit the user's specific needs. The system allows applications to write directly to the window memory buffers so that programs running on the memory-mapped console will not have to employ the slow video output that is associated with UNIX systems. $249.
Five Paces Software, Inc., 9635 Wendell Road, Dallas, TX 75243; 214/340-4933

CIRCLE 330 ON READER SERVICE CARD



Viewnix screen

A productivity package for Turbo Pascal programmers, **TurboPower Utilities**, has been introduced by **TurboPower Software**. This package includes nine high-level utilities, including a Pascal structure analyzer, an execution timer, an execution profiler, a Pascal Pretty-printer, a command repeater, a pattern replacer, a difference finder, a file finder, and a directory program. $55; with full-commented source code, $95.
TurboPower Software, 478 W. Hamilton Avenue, Suite 196, Campbell, CA 95008; 408/378-3672

CIRCLE 334 ON READER SERVICE CARD

A new addition to **Spruce Technology Corporation**'s family of syntax-directed editors has been announced: **FirsTime for Turbo Pascal**. It features automatic statement generation, automatic program formatting, full syntax checking while in the editor, and syntax-oriented cursor movements. $74.95.
Spruce Technology Corporation, 74 West Shawnee Trail, Wharton, NJ 07885; 201/663-0063

CIRCLE 332 ON READER SERVICE CARD

According to **Decision Resources**, **CHART-MASTER 6.1** and **SIGN-MASTER 5.1** are now compatible with IBM's new Proprinter and Color Jetprinter. The new versions of these programs will feature a new Helvetica-like font and a font that depicts 94 symbols of transportation, utilities, arrows, and bullets. CHART-MASTER, $375; SIGN-MASTER, $245.
Decision Resources, 25 Sylvan Road South, Westport, CT 06880; 203/222-1974

CIRCLE 342 ON READER SERVICE CARD

A number of multiuser software applications development tools have been made available by **Nestar Systems, Inc.** These tools include **MDBS III**, from Micro Data Base Systems, Inc.; **Revelation**, from Cosmos, Inc.; **LAN: Datacore**, from Software Connections; and **Btrieve/N**, from Softcraft, Inc. MDBS III is a network database management system. Revelation is a multiuser database package. LAN: Datacore is a relational applications development tool. Btrieve/N is a record manager that provides high-level access and manipulation of records in a database and that interfaces to four languages. Prices: MDBS III, $20,160; Revelation, $2,000 (four simultaneous users); LAN: Datacore, $945 (five simultaneous users per database); Btrieve/N, $595 (six users).
Nestar Systems, Inc., 2585 E. Bayshore, Palo Alto, CA 94303; 415/493-2223

CIRCLE 340 ON READER SERVICE CARD

PC WEEK'S PRODUCT OF THE YEAR
PC MAGAZINE'S AWARD FOR TECHNICAL EXCELLENCE

Borland Introduces the Laws of *TURBO DYNAMICS*™

Laws That Work Like Magic. Whether considering technological excellence, or innovation in areas such as pricing, not copy-protection, licensing agreements, site licenses, 60 day money-back guarantee—Borland is clearly recognized as the software industry leader. The following three laws of "*Turbo Dynamics*"™ exemplify our pledge for excellence.

1ST LAW:

SPEED, POWER AND PRICE.

Borland products are known to be fast, powerful and to deliver an incredible price performance ratio. We only believe in absolutely superb software at rock bottom prices.

2ND LAW:

NOT COPY-PROTECTED SOFTWARE AND REASONABLE LICENSING AGREEMENTS.

We will always offer not copy-protected versions of our software. Also, our licensing agreement is now so simple that even a child can understand it.

3RD LAW:

60 DAY MONEY-BACK GUARANTEE.

This third law is actually a first in the industry! We are so sure that you will love our software that all of our products now come backed with a 60 day money-back guarantee. No questions asked.

Turbo Dynamics Applies to Turbo Pascal. Borland's Pascal family of products is growing by leaps and bounds. You can now join hundreds of thousands of users and enter the world of Turbo Pascal programming. And remember, all three laws of *Turbo Dynamics* apply to all Borland products.

TURBO PASCAL™ $69.95

The industry standard. With more than 350,000 users worldwide Turbo Pascal is the industry's de facto standard. Turbo Pascal is praised by more engineers, hobbyists, students and professional programmers than any other development environment in the history of microcomputing. And yet, Turbo Pascal is simple and fun to use. **Free spreadsheet** included on every Turbo disk with ready-to-compile source code. ***Options:*** We offer the exciting Binary Coded Decimal (BCD) option for your business applications as well as an 8087 option for your number-crunching applications at a very low charge. Please refer to the coupon. ***Portability.*** Turbo Pascal is available today for most computers running PC-DOS, MS-DOS, CP/M-80 or CP/M-86. ***Jeff Duntemann, PC Magazine:*** "In its simplicity it achieves an elegance that no other language compiler has ever displayed."

TURBO GRAPHIX TOOLBOX™ $54.95

High resolution monochrome graphics and window management for the IBM PC. The Turbo Graphix Toolbox will give even a beginning programmer the expert's edge. It's a complete library of Pascal procedures and functions. Tools that will allow you to draw and hatch pie charts, bar charts, circles, rectangles and a full range of geometric shapes. Procedures that will save and restore graphic images to and from disk. And much, much, more. You may incorporate part or all of these tools in your programs and yet we won't charge you any royalties. Best of all, these functions and procedures come complete with commented source code on disk ready to compile.

TURBO TUTOR™ $34.95

From start to finish in 300 pages. Turbo Tutor is for everyone from novice to expert. Even if you've never programmed before Turbo Tutor will get you started right away. ***A must.*** You'll find the source code for all the examples in the book on the accompanying disk ready to compile. Turbo Tutor might be the only reference on Pascal and programming you'll ever need.

TURBO DATABASE TOOLBOX™ $54.95

The Turbo Database Toolbox is the perfect complement to Turbo Pascal. It contains a complete library of Pascal procedures that allows you to sort and search your data and build powerful applications. It's another Borland set of tools that will give the beginning programmer the expert's edge. ***Get started right away: free database!*** Included on every Toolbox disk is the source code to a working data base which demonstrates how powerful and easy to use our search system, Turbo-Access, really is. Modify it to suit your individual needs or just compile it and run. ***Remember, no royalties!***

BORLAND INTERNATIONAL

4585 Scotts Valley Drive, Scotts Valley CA 95066
Phone (408) 438-8400 Telex 172373

CIRCLE NO. 250 ON READER SERVICE CARD

NOT COPY-PROTECTED

TURBO PASCAL FAMILY

Available at better dealers nationwide. Call (800) 556-2283 for the dealer nearest you. To order by Credit Card call (800) 255-8008, CA (800) 742-1133

Carefully Describe your Computer System!

Mine is: ___ 8 bit ___ 16 bit
I Use: ___ PC-DOS ___ MS-DOS ___ CP/M 80 ___ CP/M 86

My computers' name/model is: ____________

The disk size I use is:
☐ 3½" ☐ 5¼" ☐ 8"

Name: ____________
Shipping Address: ____________
City: ____________
State: ________ Zip: ________
Telephone: ____________

60 DAY MONEY-BACK GUARANTEE

Pascal 3.0	$ 69.95	____
Pascal w/8087	$109.90	____
Pascal w/BCD	$109.90	____
Pascal w/8087 & BCD	$124.95	____
Turbo Database Toolbox	$ 54.95	____
Turbo Graphix	$ 54.95	____
Turbo Tutor	$ 34.95	____

*These prices include shipping to all U.S. cities. All foreign orders add $10 per product ordered.

Amount: (CA 6% tax) ____________

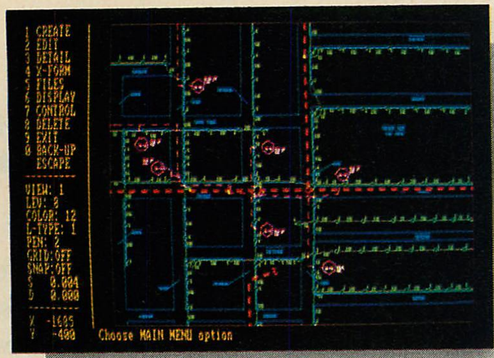
Payment: VISA MC BankDraft Check

Credit Card Expir. Date: ____/____

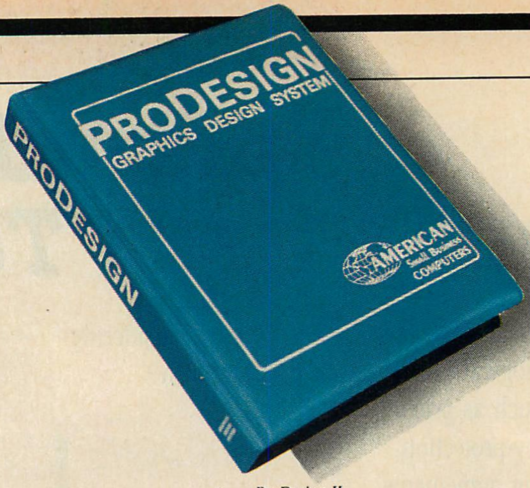
Card #: ____________

COD's and Purchase Orders WILL NOT be accepted by Borland. California residents: add 6% sales tax. Outside USA: add $10 and make payment by bank draft, payable in US dollars drawn on a US bank.

F6



CADKEY image



ProDesign II

CADKEY, from **Micro Control Systems, Inc.**, is a completely integrated, three-dimensional CAD package for the PC. With CADKEY, users can design and plot on-line or off-line at full speed at all times, on the dedicated workstation or on a minimally configured PC (128KB). No reformatting is required for different plotters. CADKEY provides immediate-mode commands, allowing users to access commonly used commands from within any other system. $1,895.
Micro Control Systems, Inc., 27 Hartford Turnpike, Vernon, CT 06066; 203/647-0220
CIRCLE 331 ON READER SERVICE CARD

Bridge Communications, Inc. has introduced a new Ethernet-to-Ethernet gateway server that permits the formation of a LAN up to 40 kilometers long—the sum of the lengths of 16 individual Ethernets. The **GS/4** (Gateway Server/4) is a high-performance internetwork router based on the Xerox Network System high-level protocols. It filters network traffic, ensuring that messages are routed only to their intended destination. $9,900 plus $150 annual software license fee.
Bridge Communications, Inc., 1345 Shorebird Way, Mountain View, CA 94043; 415/969-4400
CIRCLE 341 ON READER SERVICE CARD

A full-featured C-language development system called **Interactive C** has been announced by **IMPACC Associates**. Interactive C consists of an interpreter, a command processor, full-screen editor, source-level debugger, and execution profiler. Its multiwindow, multi-CRT user interface permits debugging of full-screen graphics programs on one or two CRTs with simultaneous display of source code, program output, and system status. $395.
IMPACC Associates, P.O. Box 93, Gwynedd Valley, PA 19437; 215/699-7235
CIRCLE 338 ON READER SERVICE CARD

WindowDOS Associates has introduced **WindowDOS**, a memory-resident program that eliminates the need to exit an applications program in order to perform disk management tasks, such as copying, erasing, or renaming a file or checking the amount of free space. WindowDOS uses 39KB of memory and requires DOS 2.0 or later. $49.95.
WindowDOS Associates, Box 300488, Arlington, TX 76010; 817/467-4103
CIRCLE 333 ON READER SERVICE CARD

WindowDOS screen

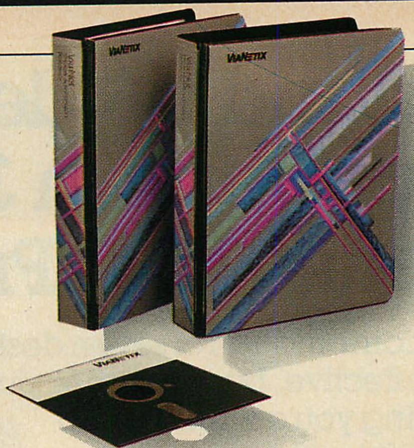
ProDesign II, a CAD package designed primarily for engineering and architectural applications, has been introduced by **American Small Business Computers**. Its drawing features include lines, curves, circles, ovals, arcs, and area fill. Sections of drawings may be moved, rotated, expanded, or reduced. User-created symbol libraries are supported, as are overlay capabilities. $299.
American Small Business Computers, 118 S. Mill Street, Pryor, OK 74361; 918/825-4844
CIRCLE 343 ON READER SERVICE CARD

dBASE software development tools with voice recognition have been announced by **Global Technologies**. The **dBASE Professional Development System** reduces the time and cost of developing new products in dBASE and gives programmers a method of storing and retrieving frequently used subroutines and programs with minimal key strokes or through voice recognition. The system is based on a specially configured version of EDIX, a programmers' editor that enables the programmer to edit and view four different command files on screen at once and to access forms graphics, macros, and subroutine libraries from within the editor. Voice recognition uses the Voice Command product from Logical Business Machines. dBASE Configured Programmers' Editor, $245; Professional Development Library, $195; Voice Command System, $895.
Global Technologies, 16572 E. Louisiana Drive, Aurora, CO 80017; 303/337-7758
CIRCLE 351 ON READER SERVICE CARD

Graphic Software Systems, Inc. has introduced the **GSS-TOOLKIT Kernel System**, which is an implementation of the ANSI and ISO Graphical Kernel System Specification Level 2b. GSS-TOOLKIT is a linkable library of graphics routines that provides program portability between computer systems. The programmer may choose among C, FORTRAN, and BASIC compilers. $495.
Graphic Software Systems, Inc., 25117 S.W. Parkway, Wilsonville, OR 97070; 503/682-1606
CIRCLE 339 ON READER SERVICE CARD

A complete Forth software development system tailored specifically for UNIX and XENIX has been introduced by **Ubiquitous Systems, Inc.** Called **u4th**, the system provides access to UNIX system calls, the ability to incorporate new primitives written in C, to compile high-level Forth words into the load image, and to pass unrecognized words through to UNIX. The object-oriented Forth extension word set is standard. It uses regular UNIX files for loading and can be run as a filter. $395.
Ubiquitous Systems, Inc., 13333 Bel-Red Road N.E., Bellevue, WA 98005; 206/641-8030
CIRCLE 348 ON READER SERVICE CARD

Pascal & Associates, Inc.



ViaNetix

Lattice, Inc. has announced the **Lattice TopView Toolbasket**, a library of more than 70 C functions to control window, cursor, and pointer operations for programmers writing applications to take advantage of IBM's TopView window environment. $250.
Lattice, Inc., P.O. Box 3072, Glen Ellyn, IL 60138; 312/858-7950
CIRCLE 344 ON READER SERVICE CARD

AGS Computers, Inc. has announced **Smart/C**, a fully integrated, precompilation development environment for the C language. Smart/C allows the programmer to create, edit, test, and debug C programs, all before any compiling is done. Its user interface is highly screen-oriented and verbose. $500.
AGS Computers, Inc., 1139 Spruce Drive, Mountainside, NJ 07092; 201/654-4321
CIRCLE 347 ON READER SERVICE CARD

The **ViaNet/UNIX** LAN from **ViaNetix, Inc.** connects UNIX and UNIX-derivative systems. Other versions of the ViaNet LAN allow DOS and UNIX computers to be mixed on the same network; DOS networking alone or with a UNIX computer acting as a file server may be selected. A virtual terminal may be added, allowing all DOS nodes to access the UNIX computer as standard terminals. Price unavailable.
ViaNetix, Inc., 5766 Central Avenue, Boulder, CO 80301; 303/440-0700
CIRCLE 349 ON READER SERVICE CARD

A software package that functionally replaces IBM's 8087 math coprocessor for use with IBM APL has been announced by **Fort's Software**. The **8087 Eliminator** is available in two versions—one for the PC*jr*, PC, and PC/XT and one that adds support for the PC/AT. $49 without AT support; $75 for the AT version.
Fort's Software, P.O. Box 396, Manhattan, KS 66502; 913/537-2897
CIRCLE 345 ON READER SERVICE CARD

RIDE, a high-level programming language for business applications, has been introduced by **Rochkind Software Corporation**. The language is designed to provide an alternative to the choice between a programming language that lacks database features and a database system, such as dBASE II. RIDE includes a built-in database system and a screen forms manager. $295.
Rochkind Software Corporation, 3080 Valmont Road, Boulder, CO 80301; 303/442-4981
CIRCLE 346 ON READER SERVICE CARD

Pascal & Associates, Inc. has released **LOCATE**, a software package that finds any word stored anywhere on a floppy- or hard-disk system in seconds. The package operates on data of any origin, including word processors and communications programs. $399.
Pascal & Associates, Inc., 136 East Rosemary Street, Chapel Hill, NC 27514; 919/942-1411
CIRCLE 350 ON READER SERVICE CARD

OTHERWARE

Data Decisions has announced a three-volume information service that provides independent hands-on evaluations for hardware and software products needed to link microcomputers to mainframes, minicomputers, data services, and other PCs. **PC Communications** provides comprehensive analysis of the entire PC communications marketplace and tests the entire range of mainstream products in this area. $800 for a one-year subscription includes the three-volume reference set, monthly updates, monthly newsletters, and unlimited free telephone consulting with micro communications specialists.
Data Decisions, 20 Brace Road, Cherry Hill, NJ 08034; 609/429-7100
CIRCLE 354 ON READER SERVICE CARD

The **X.PC** communications protocol, a public-domain standard initially developed by **Tymnet, Inc.**, is an asynchronous communications protocol that enables personal computers to handle as many as 15 channels simultaneously over a single dial-up telephone line. Microsoft's new Microsoft Access supports X.PC, as does MCIMail's electronic mail network; Concord Data Systems has announced it will integrate X.PC into its modem product line.
Tymnet, Inc., 2710 Orchard Parkway, San Jose, CA 95134; 408/942-5015
CIRCLE 355 ON READER SERVICE CARD

Microsoft Access, from **Microsoft Corporation**, is a business-information access program for electronic communications. It gives IBM PC users easy access a wide range of information services, including on-line databases, information-retrieval services, and electronic mail systems. The program comes with built-in interfaces for Dow Jones News/Retrieval, CompuServe EIS, NewsNet, and the Official Airline Guide. The product offers electronic mail with off-line editing and a programming language that lets users create their own menus or scripts; it supports both the XMODEM and the X.PC protocol. $250.

Also from **Microsoft** come new Microsoft XENIX versions of four high-level languages: **Microsoft FORTRAN**, **Microsoft Pascal**, **Microsoft COBOL**, and **Microsoft BASIC**. With these languages, programs written in a previous MS-DOS version of a language may be transported to the XENIX environment with little or no modification. In addition, programs may be downloaded from mainframes to the XENIX environment. Prices: Microsoft FORTRAN and Microsoft Pascal, $495; Microsoft COBOL, $995; Microsoft BASIC, $350.
Microsoft Corporation, 10700 Northup Way, Box 97200, Bellevue, WA 98009; 206/828-8080
CIRCLE 356 ON READER SERVICE CARD

ATTENTION SIDEKICK USERS: SUPERKEY IS SIDEKICK'S BEST COMPANION. GET SUPERKEY TODAY!

Borland's SuperKey™ lets one powerful keystroke do the work of hundreds and helps keep your confidential files . . . confidential!

SUPERKEY TURNS 1,000 INTO 1! Yes, SuperKey can *record* lengthy keystroke sequences and play them back at the touch of a single key. Instantly. Like Magic. Say, for example, you want to add a column of figures in 1-2-3. Without SuperKey you'd have to type seven keystrokes just to get started. ["shift-@-s-u-m-shift-("]. With SuperKey you can turn those 7 keystrokes into 1.

SUPERKEY HELPS PROTECT YOUR CAPITAL INVESTMENT. SuperKey, at your convenience, will make your screen go blank after a predetermined time of screen/keyboard inactivity. You've paid hard-earned money for your PC. SuperKey will protect your monitor's precious phosphor . . . and your investment. This feature alone justifies your SuperKey purchase!

SUPERKEY KEEPS YOUR 'CONFIDENTIAL' FILES . . . CONFIDENTIAL! Time after time you've experienced it: anyone can walk up to your PC, and read your confidential files (tax returns, business plans, customer lists, personal letters . . .). With SuperKey you can encrypt any file, even while running another program. As long as you keep the password secret, only YOU can decode your file. SuperKey implements the U.S. government Data Encryption Standard (DES).

SUPERKEY PROTECTS YOUR WORK FROM INTRUDERS WHILE YOU TAKE A BREAK. Now you can lock your keyboard at any time. Prevent anyone from changing hours of work. Type in your secret password and everything comes back to life . . . just as you left it.

THE CRITICS' CHOICE

"While most people only talk about low-cost personal computer software, Borland has been doing something about it. And Borland provides good technical support as part of the price."
John Markoff & Paul Freiberger, syndicated columnists

"What I think the computer industry is headed for: well-documented, standard, plenty of good features, and a reasonable price."
Jerry Pournelle, BYTE

NOT COPY-PROTECTED

Get your IBM PC or compatible a SuperKey today.

SuperKey

Available at better dealers nationwide. Call (800) 556-2283 for the dealer nearest you. To order by Credit Card call (800) 255-8008, CA (800) 742-1133

$69.95

This price includes shipping to all U.S. cities. All foreign orders add $10 per product ordered.

60 DAY MONEY-BACK GUARANTEE

YES! Please rush SuperKey to me. Send me ____ copies.

Name: ____
Shipping Address: ____
City: ____
State: ____ Zip: ____
Telephone: ____

Subtotal ____
(CA res. add 4.20 tax per copy) ____
Amount Enclosed: ____
Payment VISA MC BankDraft Check
Credit Card Exp. Date ____/____
Card # ____

COD's and Purchase Orders WILL NOT be accepted by Borland. California residents: add 6% sales tax. Outside USA: add $10 and make payment by bank draft, payable in US dollars drawn on a US bank.

K6

BORLAND INTERNATIONAL

4585 Scotts Valley Drive, Scotts Valley CA 95066
Phone (408) 438-8400 Telex 172373

CIRCLE NO. 253 ON READER SERVICE CARD

THE PROGRAMMER'S SHOP™

helps save time, money and cut frustrations. Compare, evaluate, and find products.

SERVICES
- Programmer's Referral List
- Compare Products
- Help find a Publisher
- Evaluation Literature free
- Dealer's Inquire
- Newsletter
- Rush Order
- Over 700 products
- BULLETIN BOARD - 7 PM to 7 AM 617-826-4086

Free Literature - Compare Products

Evaluate products Compare competitors Learn about new alternatives. One *free call* brings information on just about any programming need. Ask for any ***"Packet"*** or "Addon Packet": ☐ ADA, Modula ☐ "AI" ☐ BASIC ☐ "C" ☐ COBOL ☐ Editors ☐ FORTH ☐ FORTRAN ☐ PASCAL ☐ UNIX/PC or ☐ Debuggers, Linkers, etc.

RECENT DISCOVERIES

PC LINT - full C program checking and big, small model. C86, Lattice. MSDOS $ 95

ARTIFICIAL INTELLIGENCE

ARITY/PROLOG-full, debug, to ASM&C, 16 Meg use, windows, strings. With compiler $1950. MSDOS $495

ExpertEASE - Expert system tool. Develop by describing examples of how you decide. PCDOS $625

Expert LISP - Interpreter: Common LISP syntax, lexical scoping, toolbox, graphics. Compiler. 512K MAC $465

EXSYS - Expert System building tool. Full RAM, Probability. Why, serious, files PCDOS $275

GC LISP - "COMMON LISP", Help. tutorial, co-routines, compiled functions, thorough. PCDOS Call

M Prolog - full, rich, separate work spaces. MSDOS $725

PROLOG-86 - Learn fast. Standard, tutorials, samples of Natural Language. Exp. Sys. MSDOS $125

TLC LISP - "LISP-machine"-like. all RAM, classes, turtle graphics 8087 CPM-86. MSDOS $235

WALTZ LISP - "FRANZ LISP" - like, 611 digits, debugger, large programs. CPM80 MSDOS $159

MicroProlog - improved	MSDOS	$235

BASIC

ACTIVE TRACE, DEBUGGER - BASICA, MBASIC, interactive, well liked MSDOS $ 79

CADSAM FILE SYSTEM - full ISAM in MBASIC source. MSDOS $150

BASCOM-86 - Microsoft	8086	279
CB-86 - DRI	CPM86, MSDOS	419
Data Manager - full source	MSDOS	325
InfoREPORTER - multiple	PCDOS	115
Prof. Basic - Interactive, debug	PCDOS	89
TRUE BASIC - ANSI	PCDOS	125

Ask about ISAM, other addons for BASIC

EDITORS FOR PROGRAMMING

BRIEF Programmer's Editor - undo, windows, reconfig. PCDOS $195

VEDIT - well liked, macros, buffers, CPM-80-86. MSDOS. PCDOS $119

C Screen with source	86/80	75
Epsilon - like EMACS	PCDOS	195
PMATE - powerful	8086	185
XTC - multitasking	PCDOS	95

COBOL

Microsoft Version II - upgraded. Full Lev. II, native, screens. MSDOS $500

Dig Res-decent	MSDOS	525
Macintosh COBOL - Full.	MAC	1850
MBP - Lev II, native, screen	MSDOS	885
MicroFocus Prof.-full	PCDOS	call
Ryan McFarland-portable	MSDOS	695

C LANGUAGE

C-terp Interpreter by Gimpel, full K&R, .OBJ and ASM interface. 8087 MSDOS $275

INSTANT C - Interactive development - Edit. Source Debug, run. Edit to Run - 3 Secs. MSDOS $445

"INTRODUCING C" - Interactive C to learn fast. 500 page tutorial. examples, graphics PCDOS $ 95

Wizard C - Lattice C compatible, full sys. III syntax, lint included, fast, lib. source. MSDOS $450

MSDOS C86-8087, reliable	call
Lattice C - the standard	call
Microsoft C 3.0 - new	279
Williams - debugger, fast	call
CPM80 - EcoPlus C-faster, SLR	275
BDS C - solid value	125

MEGAMAX C - native Macintosh has fast compile, tight code, K&R. toolkit, .OBJ, DisASM MAC $275

MACINTOSH Hippo II	375
Consulair's MAC C, toolkit	395

Compare. evaluate. consider other Cs

C ADDONS

APPLICATION TOOLKIT by Shaw - Complete: ISAM, Screen, Overlay mgnt, report gen. Strings, String math. Source. CPM, MSDOS $395

COMMUNICATIONS by Greenleaf ($159) or Software horizons ($139) includes Modem7, interrupts, etc. Source. Ask for Greenleaf demo.

C SHARP Realtime Toolkit-well supported, thorough, portable, objects, state sys. Source MANY $600

CIndex + -full B+Tree, vari. length field. Source, no royal. MSDOS $369

dbVista FILE SYSTEM - full indexing, plus optional record types, pointers. Source, no royal. MSDOS $450

CHelper: DIFF, xref, more	86/80	135
CTree - source, no royal	ALL	369
dBC ISAM by Lattice	8086	229
Greenleaf-200 +	MSDOS	159
OTHER: C Utilities by Essential	MSDOS	129
PHACT-up under UNIX, addons	MSDOS	225
ProScreen - windows	PCDOS	275
SCREEN: CURSES by Lattice	PCDOS	125
Software Horizons - Blocks I	PCDOS	139
Turbo V - Greenleaf C, fast	PCDOS	159
Windows for C	MSDOS	175

FORTRAN LANGUAGE

MacFORTRAN - full '77, '66 option. toolbox, debugger, 128K or 512K. ASM-out option MAC $375

RM/Fortran - Full '77. BIG ARRAYS. 8087, optimize, back trace, debug. MSDOS $525

Ask about Microsoft, Supersoft, others.

MS FORTRAN-86 - Improved.	MSDOS	239
DR Fortran-86 - full '77	8086	249
PolyFORTRAN-XREF, Xtract	PCDOS	165

LANGUAGE LIBRARIES

MultiHALO Graphics-Multiple video boards, printers, rich. Animation, engineering business. Any MS language, Lattice C86 $195, for Turbo $95.

Screen Sculptor - slick, thorough, fast, BASIC, PASCAL. PCDOS $115

GRAPHMATIC - 3D, FTN, PAS	PCDOS	125
File MGNT: BTrieve - all lang.	MSDOS	215
Micro: SubMATH - FORTRAN full	86/80	250
MetaWINDOW - icons, cup	PCDOS	139
PANEL - many lang. term	MSDOS	249

OTHER LANGUAGES

ASSEMBLER-ask about Turbo ASM ($95), ED/ASM ($95) - both are fast, compatible. or MASM ($125), improvements.

BetterBASIC all RAM, modules. structure. BASICA - like PCDOS $185

SNOBOL4 + -great for strings, patterns. CPM86, MSDOS $ 85

MacASM - full, fast, tools	MAC	115
Assembler & Tools - DRI	8086	159
PL1-86	8086	495
PCFORTH - well liked	MSDOS	95

SUPPORT PRODUCTS

PLINK - a program-independent overlay linker to 32 levels for all MS languages. C86 and Lattice. $315

Multilink - Multitasking	PCDOS	265
Pfinish - Profile by routine	MSDOS	345
Polylibrarian - thorough	MSDOS	95
PolyMAKE	PCDOS	95
ZAP Communications-VT100. TEK 4010, full xfer	PCDOS	65

DEBUGGERS

Periscope Debugger - load after "bombs", symbolic, "Reset Box", 2 Screen, own 16K. PCDOS $279

Advanced Trace 86 Symbolic	PCDOS	149
Atron Debugger for Lattice	PCDOS	395
CODESMITH-86 - debug	PCDOS	129
PFIX-86 Debugger	MSDOS	169
TRACE86 debugger ASM	MSDOS	115

Call for a catalog, literature, and *solid* value

800-421-8006

THE PROGRAMMER'S SHOP™

128-P Rockland Street, Hanover, MA 02339
Mass: 800-442-8070 or 617-826-7531 8517

Note: All prices subject to change without notice. Mention this ad. Some prices are specials. Ask about COD and POs. All format's available. UNIX is a trademark of Bell Labs.

TECH NOTEBOOK 45

JOHN WALKENBACH

Printer Control

SideKick's Notepad function and Spotlight's Filer conveniently send printer control characters from any program.

Desk-top utility programs such as Borland International's SideKick and Software Arts' Spotlight afford PC users a new level of convenience. These programs function as calculators, notepads, telephone dialers, appointment calendars, and alarm clocks. Once loaded they are memory-resident, ready to be called up at the touch of a key at any time—even while another program is running.

Now these programs have displayed yet another capability: they can be used to send control characters to a printer for special effects such as condensed type or bold face.

Imagine that a user is running his favorite compiler, and is ready to produce a listing of his program. But he realizes that the line length of the listing is more than 80 columns, so he will need to use condensed type. Normally, he would have to exit the program, get back to DOS, and issue the proper commands to set up his printer for condensed type. Many people do this using short BASIC programs or specially designed utility programs. The technique described below permits the user to send these control characters—at any time—using SideKick's Notepad function.

The user's first step is to set up a file containing the printer control codes (with brief explanations) that he finds most useful. SideKick's Notepad can be used to create this file. These control codes usually consist of an escape character followed by one or more additional characters. For example, on the IBM Graphics Printer, the sequence Esc-E puts the printer into emphasized print mode. Esc-F cancels emphasized mode and returns the printer to normal print.

Figures 1 and 2, respectively, show control files for the IBM Graphics printer and for a Gemini 10X printer. The left column contains the escape sequences; the right column describes what they do. To send an escape sequence to the printer, call up SideKick as usual with Ctrl-Alt. Then type N for the Notepad function, followed by F3 for new file. At the prompt, type the name of the printer control file. Move the cursor down the left column to the function desired. Next, mark a block that contains only the escape codes. Use Ctrl-K Ctrl-T, SideKick's command to mark a word. Finally, send the marked block to the printer with the Ctrl-K Ctrl-P.

The printer control file will vary for different printers. SideKick's ASCII table function may be useful for converting the codes in any printer manual to keyboard codes.

Setting up the printer control file may not be as easy as it sounds. For instance, an attempt to input an Esc character in SideKick's Notepad will end SideKick. Further, many of the codes needed are SideKick commands, so they cannot be entered directly. Avoid this by preceding keyboard control characters with Ctrl-P, SideKick's control character prefix.

To control the printer with Spotlight, each command sequence must be put in a separate file: one file should contain the command sequence to start compressed printing and another file should contain the command sequence to end compressed printing. Using Spotlight's Filer, each file is copied to the printer as needed. In this case, since the entire file is sent to the printer, only the command sequence should be put into the file; comments should be omitted.

John Walkenbach, Ph.D., is vice-president of marketing for a large credit union located in Los Angeles, California. He has more than 15 years of experience working with computers.

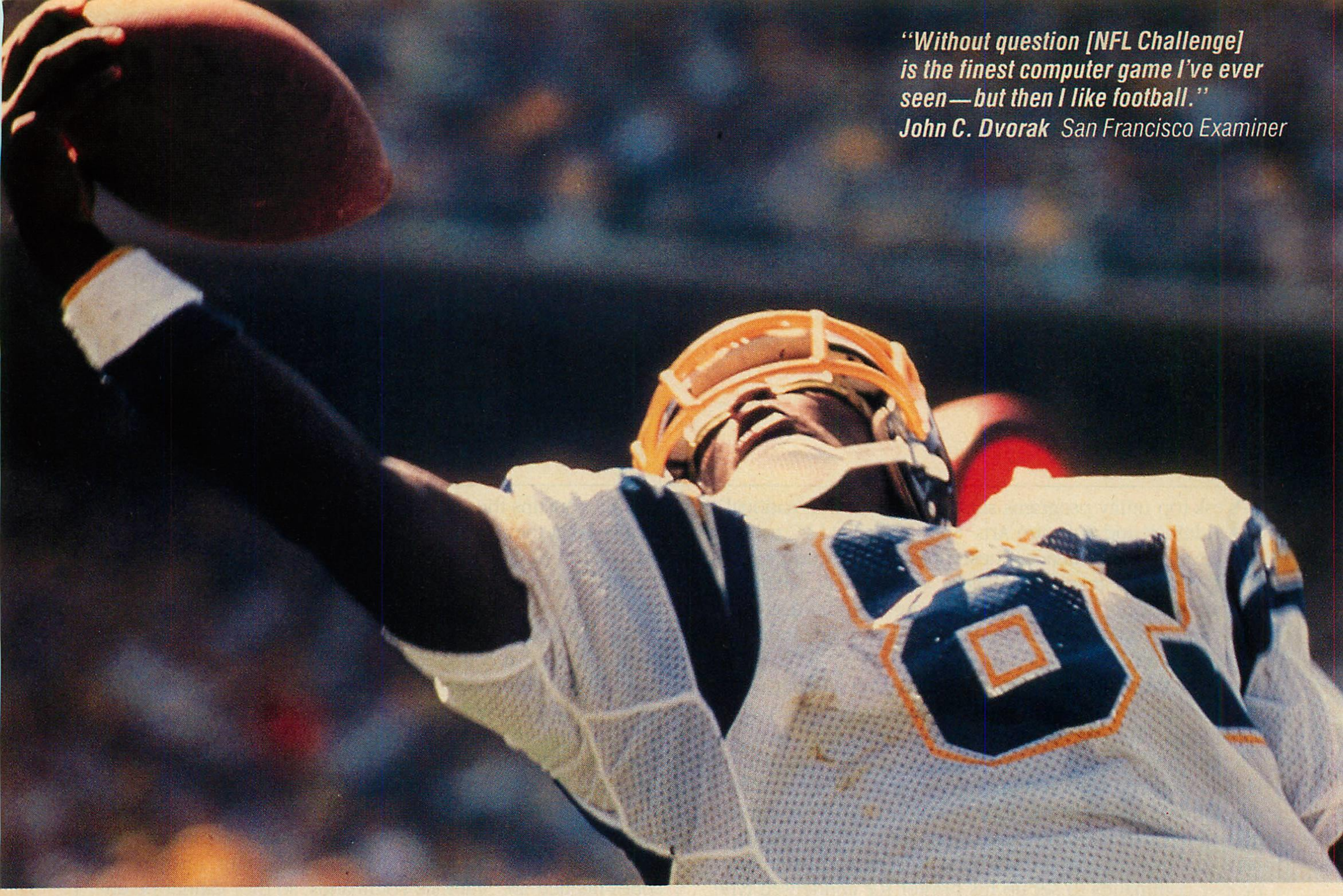
FIGURE 1: *Control File for Graphics Printer*

ESCAPE SEQUENCE	PRINTER FUNCTION
[W1	Double width mode on
[W0	Double width mode off
^O	Compressed mode on
^R	Compressed mode off
[G	Double strike mode on
[H	Double strike mode off
[E	Emphasized mode on
[F	Emphasized mode off
[-1	Underline mode on
[-0	Underline mode off
[S1	Subscript mode on
[S0	Superscript mode on
[T	Cancel Subscript/Superscript mode

FIGURE 2: *Control File for GEMINI 10X Printer*

ESCAPE SEQUENCE	PRINTER FUNCTION
[4	Italic character set
[B^A	Pica (10 cpi)
[B^B	Elite (12 cpi)
[^O	Condensed (17 cpi)
[^R	Cancel condensed
[G	Double strike mode on
[H	Double strike mode off
[E	Emphasized mode on
[F	Emphasized mode off
[-^A	Underline mode on
[-^@	Underline mode off
[S^@	Superscript mode on
[T	Superscript mode off
[@	Initialize printer

The left bracket in the above figures represents the escape character. For clarity, control characters are shown here preceded by a caret (^). In SideKick, these characters should be entered using Ctrl-P; the caret will not appear in the file.



Put the NFL at your fingertips.

Hands-on pro football excitement is as close as your keyboard with **NFL Challenge** from Xor. This is the officially licensed NFL action computer game with offensive plays and defensive sets based on the ones in NFL playbooks, plus complete updatable rosters for all 28 NFL teams.

Every game can be the Super Bowl when you take the field with Xor **NFL Challenge** software. Feel the pressure as you choose starting lineups, drive downfield against the clock, and go head-to-head, down-by-down against the league's most innovative offenses and formidable defenses. It's the most intense computer simulation of the pro football experience ever devised.

Xor's **NFL Challenge** requires IBM® PC or PC/XT with DOS 2.0, 256K bytes of memory, monochrome monitor with IBM® monochrome card or RGB color monitor with IBM® color graphics adapter. Also runs on the IBM® PC/AT.

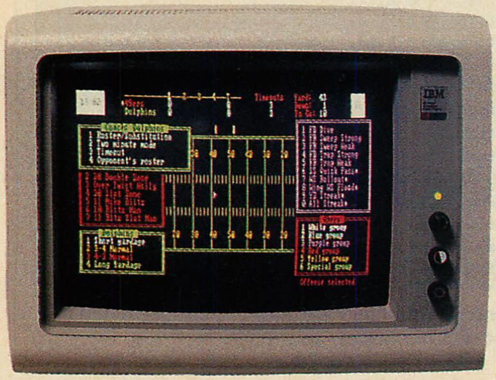
With **NFL Challenge** from Xor, you have 50-yard line seats for every game!

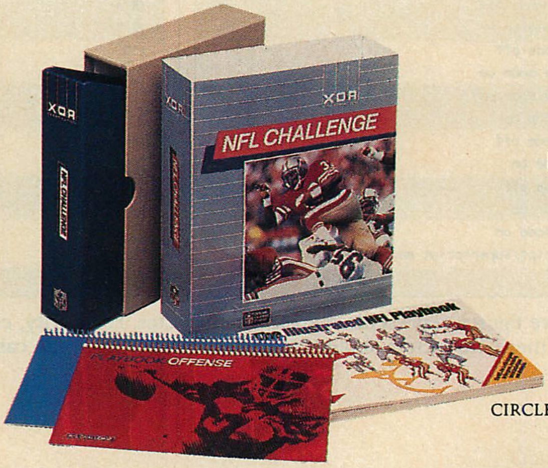
Ordering Information

NFL Challenge will be in short supply. To kick off your NFL season, contact us for the name of the dealer nearest you:

1-800-328-8322 ext. 538
(in MN call (612) 938-0005)

NFL Challenge will be reserved for you. Product will be available in limited quantities beginning in August, 1985.





XOR™
CORPORATION
5421 Opportunity Court
Minnetonka, Minnesota 55343
(612) 938-0005

NFL Officially Licensed Product

CIRCLE NO. 228 ON READER SERVICE CARD

JEFF DUNTEMANN

Untyped VAR Parameters

Turbo Pascal permits passing a parameter for which the type is unknown at compile time.

Of all the ways in which Turbo Pascal has enhanced the inept ISO standard for Pascal, none is quite so bizarre as its notion of untyped VAR parameters. A procedure parameter declared by reference (that is, with the VAR prefix) may be declared without a data type for the parameter: the user can plug a variable of any type into the parameter without so much as a peep from the compiler.

How does this work? When a procedure with a VAR parameter is called, the calling logic passes the procedure only the *address* of the actual parameter. An 8086 address is always four bytes (segment : offset) no matter what is located at the address, so the type of the parameter is of little concern in *passing* the parameter.

Processing the parameter is another matter. Program code must know the type of a parameter in order to do anything with it. A parameter's type contains all available information about how large a parameter is and how its data are arranged. Theoretically, a procedure should be powerless to work with an untyped VAR parameter. And that is, in fact, the case: an untyped VAR parameter is incompatible with all other types. The user cannot assign anything to it, nor can he assign its value to anything.

Although nothing is known about the size and structure of an untyped VAR parameter, the location of the actual parameter is known, since an untyped VAR parameter is the machine address of the actual parameter passed to the untyped VAR parameter at runtime. Using Turbo Pascal's ABSOLUTE key word, another variable can be declared *on top of* the actual parameter, and the untyped parameter accessed through the absolute variable. If the absolute variable is a large array of bytes, the untyped actual parameter can be viewed on a byte-by-byte basis by examining the array of bytes that occupies the same physical memory space. The code that uses the ABSOLUTE array should know the physical size of the untyped parameter.

The procedure below (VARDUMP.PAS) is a debug tool that permits a hex dump of *any* variable at runtime, regardless of type. The user calls

VARDUMP (CON, <variable>, SIZEOF(<variable>));

for a screen dump or CON can be replaced with LST for a printer dump. The ITSIZE parameter tells how much of the absolute array, DUMPIT, overlaps the actual parameter passed in TARGET. The untyped VAR parameter is treated as a series of bytes, which are dumped to the screen in hex format, 16 to a line. The actual parameter passed to VARDUMP in TARGET may be from one byte to MAXINT (32,767) bytes.

Procedure VARDUMP also includes a small, but generally useful, procedure WRITEHEX, which will display or print a byte in hexadecimal format.

Untyped VAR parameters are the only effective way to pass a parameter for which the type is unknown at compile time. Dumping the contents of a variable for debugging is just one application. The untyped VAR parameters also can be used to sort arrays for which the type and size are not known at compile time. Bizarre—but useful!

LISTING: VARDUMP.PAS

```
PROCEDURE VARDUMP(VAR DEVICE : TEXT;
                  VAR TARGET;
                  ITSIZE : INTEGER);

CONST PRINTABLES : SET OF CHAR = [' '..'}'];

VAR I,J         : INTEGER;
    FULL,LEFT : INTEGER;
    DUMPIT    : ARRAY[0..MAXINT] OF BYTE ABSOLUTE TARGET;

PROCEDURE WRITEHEX(VAR DEVICE : TEXT; BT : BYTE);
CONST HEXDIGITS : ARRAY[0..15] OF CHAR = '0123456789ABCDEF';
VAR BZ : BYTE;
BEGIN
  BZ := BT AND $0F;
  BT := BT SHR 4;
  WRITE(DEVICE,HEXDIGITS[BT],HEXDIGITS[BZ])
END;

PROCEDURE DUMP_LINE(OFFSET,BYTE_COUNT : INTEGER);
VAR I : INTEGER;
BEGIN
  FOR I := 0 TO BYTE_COUNT-1 DO   { Hex dump the data }
    BEGIN
      WRITEHEX(DEVICE,DUMPIT[(OFFSET*16)+I]);
      WRITE(DEVICE,' ')
    END;
  FOR I := 0 TO 56 - (BYTE_COUNT*3) DO WRITE(DEVICE,' ');
    WRITE(DEVICE,'|');             { Show first boundary bar }
  FOR I := 0 TO BYTE_COUNT-1 DO   { Show printable equivalents }
    IF CHR(DUMPIT[(OFFSET*16)+I]) IN PRINTABLES THEN
      WRITE(DEVICE,CHR(DUMPIT[(OFFSET*16)+I]))
    ELSE WRITE(DEVICE,'.');
  WRITELN(DEVICE,'|')              { Final boundary bar }
END;

BEGIN
  FULL := ITSIZE DIV 16;   { # of 16-byte chunks in TARGET }
  LEFT := ITSIZE MOD 16;   { # of 'leftover' bytes }
                     { Hex dump all full 16-byte chunks: }
  FOR I := 0 TO FULL-1 DO
    DUMP_LINE(I,16);
                     { If TARGET is shorter than 16 bytes or }
                     { if leftovers remain, dump: }
  IF LEFT > 0 THEN
    DUMP_LINE(FULL,LEFT);
  WRITELN(DEVICE)          { Space down one line after dump }
END;  { VARDUMP }
```

For everyone who ever tried doing five things at once

The perfect computer program for someone as busy as you. It lets you keep several other programs working at once.

Do you ever go in so many directions so fast not even a computer can keep up with you?

Well, now an IBM Personal Computer can—thanks to IBM TopView.

TopView is a new kind of software that lets you switch between other programs as quickly as you can change your mind, even run several programs at the same time.

Once you load TopView into your computer, you load the other programs you use most—as many as your computer's memory will permit.

After that, the greatest distance between two programs is just a couple of keystrokes, or (optional) mouse moves.

There's no waiting and a lot less diskette swapping.

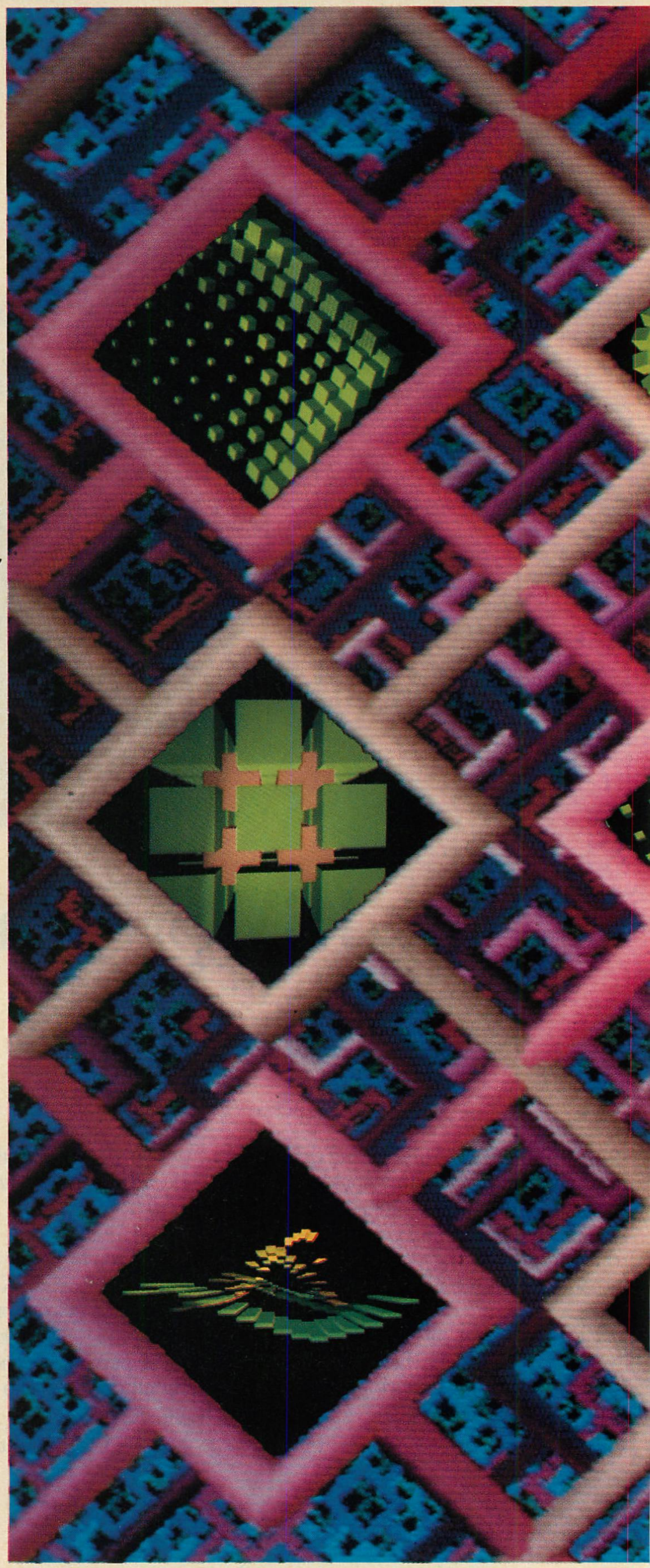
But when you're *really* busy is when TopView really shines, letting you do many jobs simultaneously.

For example, you can print a letter, while you search a file, while you analyze a spreadsheet, while your clock/calendar reminds you that your automatic dialer is about to place a call for you.

Little Tramp character licensed by Bubbles Inc., s.a.


TopView
IBM

Evaluating Data Managers as Development Tools



Here begins our long-range search for the ideal data manager designed for the technically sophisticated developer.

JULIE ANDERSON

Finding and evaluating a data management product is not an easy task. Data managers are complex applications development systems that are not easily mastered. Exploring the subtleties and uncovering the powers of each package takes an enormous investment in time.

We at *PC Tech Journal* are investing that time to uncover the ideal data management product for our readers. Our goal is to find the data manager that best answers the needs of the technically sophisticated developer. We recognize that the developer needs a tool with which he can create a custom, full-featured application for the nontechnical end user. With this in mind, we spent several months developing the criteria upon which we will judge all data management products in our long-range series. The first article reviews Data Access Corporation's DataFlex; written by Chris Christian, it begins on page 52, following this introduction.

COMPUTER GRAPHIC • DOV JACOBSON

A data manager is often thought to be simply a means to store and organize data. A common metaphor used is that of an electronic filing cabinet. Filing, however, is only one aspect of a true data manager's job. It must also be able to present the data upon demand in a variety of formats and from a variety of viewpoints and to perform basic operations and calculations on the data.

This capability in a data manager allows the user to analyze the data and make decisions based upon his observations. It moves the data manager beyond the tired metaphor of the substitute filing cabinet and into the realm of office productivity tool. Complete vertical applications with complex data relationships can be developed. These powerful data managers are the targets of our long-range series of reviews. We will review them within the context of their suitability as applications development tools.

We have developed what we believe is the ultimate specification for reviewing data management products from the applications developer's perspective. We have compiled a list of the features that are essential to building any application and have supplemented it with those functions that, while not necessary, are useful for particular types of applications.

To ensure a sufficiently in-depth examination of each product, we will review one product at a time instead of offering a single all-inclusive review of many products. Our reviews will be written according to the same criteria and will be presented in a consistent format. As the series grows, *PC Tech Journal*'s readers will have a body of individual reviews that can be used to compare one data management product against any of the others.

Each article will conform to a standard outline. The first section will reflect the purpose best served by the product, its strengths, the applications for which it is best suited, and what distinguishes it from other data managers currently available. Supplementing this section will be a summary of features, supplied by Data Decisions (our sister organization). This summary, based on Data Decisions' research report on the individual product, will provide an overview of the product so that readers will be able to refer quickly to its capacities and specifications.

The heart of each article will appear in the second section: a detailed look at the features and capabilities of each data manager. We will focus on the design of the database—the model

SAMPLE APPLICATION SPECIFICATIONS

To exercise the features *PC Tech Journal* seeks in a data management package, we designed a sample editorial inventory application. It will be implemented by each reviewer in this series examining data managers.

Fields for each file have been selected so that they define a variety of data types and editing specifications. These are described in tables 1, 2, and 3 and include field type, size, and editing specifications. The sample full-screen data entry form that is to be implemented is shown below; it has been designed to combine data from multiple files on a single form.

Reports. The product should be able to handle the following reports.

- A columnar report is to be generated that lists for each issue the article titles, author names, number of editorial pages, and number of listing pages per article. The data will be sorted by issue booked (volume and number). Subtotals per issue will be calculated for editorial page count and listing page count and a total given for all pages. At the end of the report, a grand total should be calculated for all pages. The issue volume and number will be printed at the beginning of the section containing its articles. The headers and footers to be printed for each page will include the report title, time and date of the report generation, and the page number. This report must be run either for the data in the entire file or for a manually selected subset of records.
- Mailing labels are to be printed for all authors in an issue. Duplicates should not be printed.
- A report, sorted by author, is to be generated that prints all article titles, booking information, and fee paid, including bonus, with a total given for each author per year.

Queries. The following questions could arise in daily use and the data manager's query facility should be able to answer each one with no trouble.

- How many pages are booked in an issue? This should be divided into the number of editorial pages and the number of listing pages, with a combined total given.
- Which articles are booked for a particular issue?
- How much was paid for all articles combined in a particular issue? This number should include payments plus bonuses.
- What is the average fee that is paid to authors?
- How many pages in a given issue were devoted to product reviews?
- How much was paid per printed page in an issue? This should include only articles that are not regularly appearing departments.
- Which articles were received after the deadline for an issue?

Benchmarks. All of the benchmark tests are to be performed on the author file because it contains the largest number of records. Each set of tests will be run in single-user mode on a PC/AT in a freshly formatted fixed-disk partition using DOS 3.0. Measurements will reflect processing time only, not set-up time.

- Nine-hundred records are to be added to the empty author database from a delimited-ASCII format file.
- An index is to be created on the state field followed by zip code.
- A list of all unique state codes is to be created in alphabetical order.
- Each occurrence of the state code CO is to be changed to CL.
- All authors who live in California are to be selected, sorted by zip code, and the records written to an external file in delimited-ASCII format.

—*JA*

```
                  EDITORIAL INVENTORY -- Article File

      Title:  ..............................................................
     Author:  ..................  ............  Day Phone:  ***/***-****
  Co-author:  ..................  ............

    Booking:  Volume .  Number ..
       Date:  ********* ****

Article Type:   Category:  .................
              Department:  .....................

   Date due:  ********            Date Received:  ........

      Sizes:  Editorial:  ..      Payments:  Article:  ....
               Listings:  ..                   Bonus:    ..
                  Total:  ++                   Total:  ++++
```

TABLE 1: *Sample Author File*

FIELD	TYPE	SIZE	EDITING SPECIFICATION
Last Name	A	18	Required entry.
First Name	A	12	Required entry.
Address	A/N	20	
City	A/N	16	
State	A	2	Must be a valid state.
Zip	A/N	5	Must be numeric.
Work Phone	A/N	10	Must be numeric.
Home Phone	A/N	10	Must be numeric.
Social Security Number	A/N	9	Must be numeric.
Biography	Note	200	Variable length text field, if possible.

The Author file, containing 900 records, is indexed on Last Name followed by First Name. The author Last Name and First Name fields link records in the Article file to the Author file. Editing must be performed for required entries and sets of valid values. The length of the file will be affected by the data manager's handling of the variable-length Biography field.

TABLE 2: *Sample Article File*

FIELD	TYPE	SIZE	EDITING SPECIFICATION
Volume	N	1	This field matches the volume field in the Issue file.
Number	N	2	This field matches the number field in the Issue file.
Category	A	17	Must be one of the following values: Product Review, Technical Article, Department, or Tech Notebook.
Department	A	21	If category is Department, then this field must be one of the following values: Programming Practices, Directions, Legal Brief, or Product of the Month. If category is not Department, then this field must be blank.
Title	A/N	60	Required entry.
Author Last Name	A	18	This field matches Last Name in Author file.
Author First Name	A	12	This field matches First Name in Author file.
Co-author Last Name	A	18	This field matches Last Name in Author file.
Co-author First Name	A	12	This field matches First Name in Author file.
Commissioned	Logical	1	Must be True if the article was commissioned or False if it was unsolicited.
Date Received	Date	8	
Editorial Pages	N	2	
Listing Pages	N	2	
Payment	N	4	
Bonus	N	2	Minimum value \$5; Maximum value \$20.

The Article file, containing 360 records, is indexed on Volume followed by Number. Volume and Number link the Article file to the Issue file, and the author name fields link records to the Author file. Editing of these fields requires the look-up of valid values in another file and editing of the Department field based on the value of the Category field.

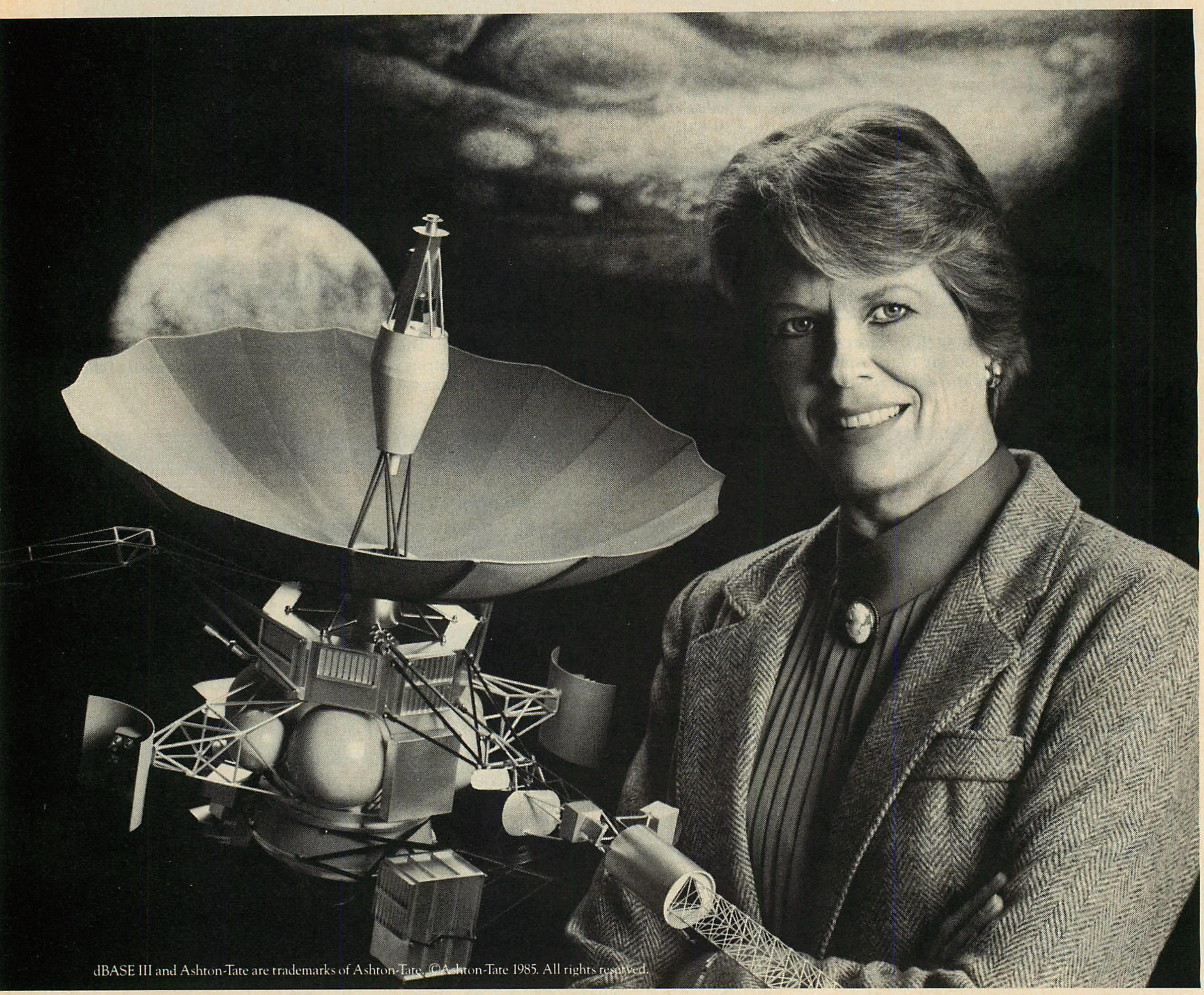
TABLE 3: *Sample Issue File*

FIELD	TYPE	SIZE	EDITING SPECIFICATION
Volume	N	1	
Number	N	2	Must be between 1 and 12 inclusive.
Deadline	Date	8	This is the deadline for receiving articles for this issue.
Month	A/N	9	
Year	A/N	4	

Month and year are optional fields. If there is a look-up facility provided by the data management system, it can be used and the fields deleted from the file. If no such facility is provided, then these fields should be included in the file to store the month and year corresponding to the issue volume and number. Volume 3 is 1985; volume 4 is 1986; volume 5 is 1987. Number 1 is January, number 2 is February, and so on.

The Issue file, containing 36 records, is indexed on Volume followed by Number. These fields link the Issue records to the Article file. Month and Year can be look-up fields if this feature is supported by the data manager. The Deadline field combined with the Date Received field of the Article file provides an opportunity for performing date arithmetic.

NASA has one shot to Jupiter. They'll go with dBASE III.



Karen Boyle,
Data Programming Coordinator,
Project Galileo

NASA has only one shot at Jupiter with Project Galileo, so there's no room for error. The purpose of the project is nothing less than to find out the origin of the solar system and seek the answer to the nature and origin of life itself.

Galileo's success is all in the timing, and that's where dBASE III™ from Ashton-Tate™ takes control. dBASE III tracks the details of the sequence of launch events.

dBASE III was easy to bring on board. It's powerful (one billion records) and has a built in programming language that has been taught to speak "Galileo." dBASE III deals with the complex test details for Galileo as easily as it will deal with the complex details of your business for you.

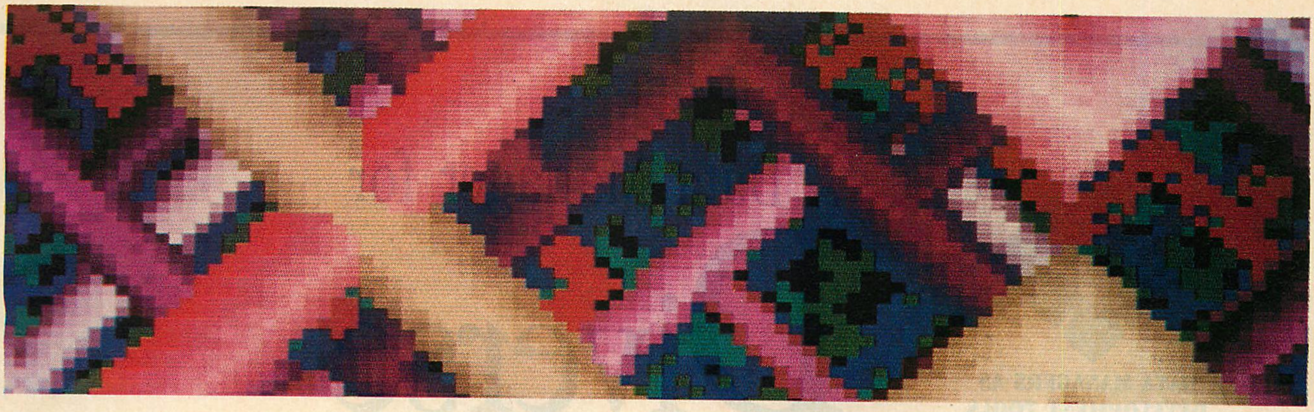
When you've only got one shot at getting something right, you need the most powerful and popular data management system on Earth ...or Jupiter.

For a dealer near you, call (800) 437-4329, ext. 2330. In Colorado call (303) 799-4900, ext. 2330.

dBASE III. The data management standard.

ASHTON·TATE
SOFTWARE

CIRCLE NO. 109 ON READER SERVICE CARD



used and the data types supported. In realizing that databases are dynamic structures, we will discuss how easily a field may be added to or deleted from a file, or have its length changed.

A data manager provides a development environment. Because of this, we will examine the products as if they are programming languages. We expect to find looping structures, local variables, subroutines with parameter passing, functions for string processing, and interfaces to other languages. We also will look for a program editor, debugging support, and interfaces to DOS functions. Whether the language is compiled or interpreted is important, and the presence of a runtime package can protect the developer's code and can lower distribution costs.

Even though we assume the developer is technically competent, we recognize the value of applications generators and other utilities for file creation and screen definition. These aids reduce the development chore by generating a core of file maintenance code that allows the developer to concentrate on writing the specific application.

We will ask how much help is provided to the developer in the area of data protection and security. Is a data dictionary present with the editing criteria for each field? What types of data validation does the data manager perform automatically and what types must be programmed? Can unauthorized access to the database be restricted?

How is data integrity assured? Are utilities provided to restore damaged files? Are checkpoint/restart capabilities provided? Can transactions be logged, and if necessary, backed out?

As more local area networks are installed, the demand for multiuser support will increase. The data manager in the LAN environment must provide record locking with deadlock detection. Moreover, records must be locked for as brief a period of time as possible so as not to deter performance.

The degree to which the product provides support for the end user is an important measure of its effectiveness. The data manager must make intuitive use of the IBM PC keyboard: it should respond to the cursor movement keys and use a consistent keyboard interface throughout the application. The nontechnical end user should be able to make ad hoc use of the system. He should be able to make interactive, nonprogrammed queries of the database outside the scope of the application. If the end user is required to know file structure or field names, the data manager must be able to display that information.

The reviewer will discuss how the product lives up to its intended purpose, how reliable it is, and how suitable the product is for applications development.

We will examine the facilities provided for the import and export of data. Getting data into and out of the data manager's files is often an overlooked capacity, but it is important for more than the initial loading of the file. Moving information between the data manager and document or spreadsheet files is necessary to achieve integration.

For those data managers that are microcomputer versions of mainframe applications, we will examine compatibility of features and interchangeability of data formats and program code.

As part of the review process, a substantial sample application will be implemented using each data management product. Each reviewer will build the editorial inventory system described in the accompanying sidebar using sample data supplied by *PC Tech Journal*. The application has been designed to exercise the features that we define as necessary in a data manager. Each author will discuss the joys and the problems encountered in developing the sample application on the product he reviews and the time and effort required in terms of programming.

Good performance is essential to a data manager, especially when large databases are being manipulated. To provide some common metrics upon which the performance of each product can be compared, five benchmarks will be run. The benchmarks are designed to measure the data manager's performance in data retrieval (including indexed and sequential), adding new records, updating data, and indexing. The reviewer will comment on product design features that affect performance and which methods a developer can use to tune his application.

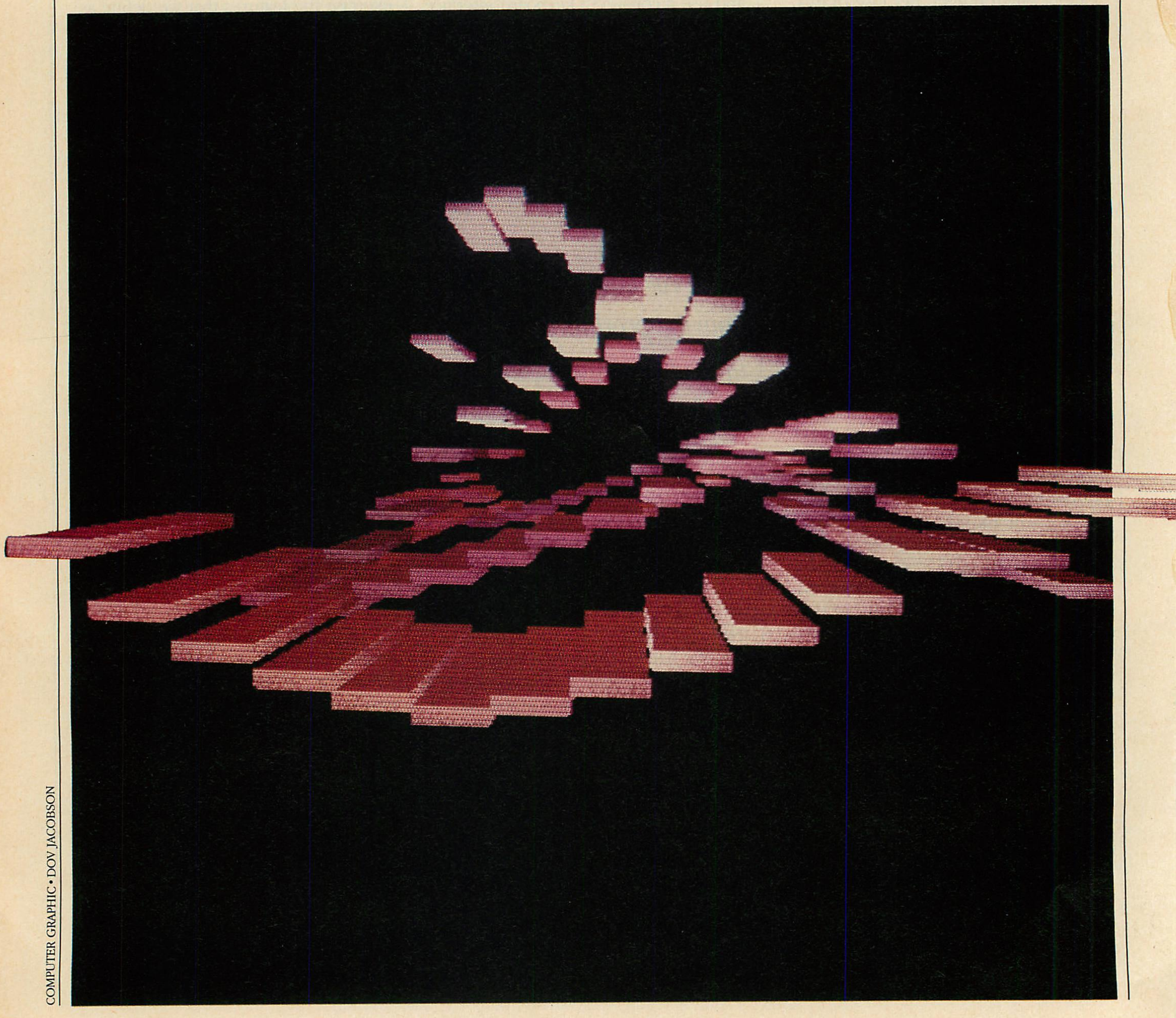
Finally, a subjective assessment will be made. The reviewer will discuss how the product lives up to its intended purpose, how reliable it seems to be, and how suitable the product is for applications development.

In future issues we will review R:Base 5000, ZIM, KnowledgeMan, Revelation, Informix, PC/Focus, and others. Sorting through the multitude of data management products is a formidable task, but we believe that this series of reviews will give *PC Tech Journal* readers the in-depth information they require to choose the package that best fits their needs as developers.

Graphics for this series on data managers, including the illustrations for this and the following article and the cover, were produced using Artwork, a software package developed by West End Film, 157 Pelton Avenue, Staten Island, NY 10310.

DATA MANAGERS AS DEVELOPMENT TOOLS

A Data Manager for Diverse Environments



COMPUTER GRAPHIC • DOV JACOBSON

Data Access Corporation's DataFlex affords applications developers versatility and operating-system portability.

CHRIS CHRISTIAN

DataFlex combines effective multi-database management with a fourth generation language to meet the needs of microcomputer applications developers. Data Access Corporation designed DataFlex to provide a solution to business problems that are data and transaction oriented. Generally, DataFlex is not well-suited to applications requiring word processing, spreadsheet analysis, or higher level math functions. What it does provide is a versatile means to data management with built-in compatibility and portability to other operating systems.

Of course, DataFlex runs on PC-DOS, but it is especially designed for multiuser operating systems including the IBM Network, Novell NetWare, Corvus PC, 3Com EtherSeries, PC-Net, Concurrent CP/M-86, and others. All 8- and 16-bit versions are written in Digital Research's Pascal MT+. A C language version has been implemented for the XENIX operating system. DataFlex is compatible with most all Intel-based, 8-bit, single and multiuser operating systems as well. One of the main attractions of DataFlex to a software developer is that a program compiled to run with one operating system will work without change on any of the others. And a program written for a multiuser environment works equally well in a single-user environment.

The multi-environment/single source-code approach of DataFlex allows developers to address very broad markets for specialized applications. A vertical market application can thus grow with a customer from a single-user 8-bit machine to a 16-bit machine, and then later to a network or UNIX-like system. For each of these systems, DataFlex can operate as a functional shell, insulating the end user from the peculiarities of the operating system. The customer's investment in training and procedures, therefore, is financially protected as computing requirements begin to grow.

A dual floppy disk IBM PC with 256KB is sufficient to develop DataFlex applications. If additional memory is present, DataFlex will use it, and a fixed disk makes development more convenient. Runtime applications can require as little as 128KB.

As shown in the accompanying sidebar, compiled from material supplied by the Data Decisions reference service, DataFlex is priced separately for development and runtime systems and for single and multiuser operating systems. Those developers who want to customize DataFlex or more closely couple their applications with its inner workings can purchase DataFlex unlinked and in library form. The latter comes with documentation for the library and some source code. Quantity discounts are available when 10 to 500 licenses are purchased.

Since its introduction in 1981, DataFlex has been shipped to more than 19,000 sites, ranging from single-person OEM shops to internal consultants for Fortune 500 companies. Three out of every four licenses are purchased for 16-bit systems, and licensing for single and multiuser systems is split fifty-fifty. Data Access provides telephone support and establishes working relationships with many developers.

DATABASE DESIGN

DataFlex is a file-handling system that works with a runtime system. Support utility programs supplement the built-in functions, yielding a full-featured applications development system. It is very sophisticated in many of its approaches to solving common business problems.

Files are composed of fixed-length records. An indexed sequential access method with a B-tree structure is used for multikey indexes. DataFlex programs are written in the DataFlex language and compiled into pseudocode that drives the runtime program.

DataFlex's capacities are generous. Each file can have up to 255 fields totaling 16KB per record. Each file can logically accommodate 16.7 million records for a total file size of two gigabytes. ASCII fields can be up to 32KB per field. Numeric fields can have 14 places to the left of a decimal point and 4 places to the right. Numeric data are stored in packed BCD ranging in size from one to three bytes. Date fields always occupy three bytes and internally store the month, day, and four-digit year. Arithmetic operations can be performed on date values. Thus, a user can easily create an aged accounts receivable report showing customers with payments 30, 60, and 90 days past due.

Each data file has an associated file definition that serves as a limited data dictionary and facilitates some program/data storage independence. The file definition itemizes for each field the field number, the field's tag name, its offset within the record and its length in bytes, the field type (ASCII, numeric, or date), the number of decimal points for numeric fields, the main index number for the field, the number of a related file and the field number within that file. The tag name is used within the QUERY utility and within DataFlex programs to refer to the fields by name rather than number. One interesting aspect of the file definition is the ability

to have multiple definitions for the same area within a record. Called an *overlap* field, this area could have an alternate name specified for a part of a field or for contiguous fields. This is useful in overcoming what would otherwise be limitations of the DBMS. Because many DataFlex functions require the knowledge of a file's number as well as field numbers, the file definition is an essential reference during an application's development.

A file's number is established at the time it is defined. DataFlex files are defined and maintained with the FILEDEF utility program, which uses an umbrella *configuration file* that contains, along with the registration information and program behavior options, a list of the currently defined files. DataFlex accommodates 255 files per configuration in this manner. From the applications developer's point of view, however, a separate configuration file is required for each application. DataFlex does not support DOS 2.0's subdirectory paths, which, on a hard disk, virtually requires all DataFlex programs and files to be within the current directory; this may be difficult to manage.

The benefit of using file numbers and field numbers is that data relations can be maintained internally at very high speeds during file maintenance, reports, and queries. The creators of DataFlex have made some very deliberate trade-offs between end-user application speed and developer ease of use.

Each database thus has several components: a permanent entry in the configuration file that provides the file ID number; a file-definition file (.FD extension); a file containing the field names (.TAG extension); and the data (.DAT extension). Each database index has its own file (.K*n* extension, where *n* is the index number).

Each data file can have as many as nine permanent indexes plus one ad hoc index. Each index can be made up of from one to six fields. DataFlex has no provision for index *expressions*. This is a deficiency. In designing an index to a relatively short file, say 300 records, by last name, it is often useful to base the index on only the first six letters of the last name rather than an entire 24-character field. When this is done, the index file itself is one-fourth the size of a full index, and by adding a little programming, an exact match can be determined with the applications program rather than with the DBMS. A smaller index works faster.

DataFlex does, however, give the enterprising developer a way to achieve this benefit. The overlap field type is used to define a field starting at any position within the data record and continuing for any length. The developer can simply define the overlap field as starting at the last-name position and continuing for six characters. Making this the index for the file puts everything into place. In situations in which a record requires more than six fields for an index, an overlap field can be defined to span several fields, provided the fields are adjacent and in order.

Two other aspects of indexes need to be mentioned. First, DataFlex is based on an indexed sequential access method (ISAM) that requires each entry in the index structure to be unique. The solution that forces uniqueness is to add the record number field (field 0) to the index when apparent duplicates are allowed within the application. AUTODEF does this automatically if the user specifies that the indexed field will not contain unique values, but the developer must be aware of this when maintaining the file's indexes through REINDEX or FILEDEF.

Second, DataFlex does not use the ASCII collating sequence (which is 123 . . . ABC . . . abc . . .). Instead

DataFlex does not support DOS 2.0's subdirectory paths, which, on a hard disk, virtually requires all DataFlex programs and files to be within the current directory.

it uses a *dictionary* sequence of AaBbCc . . . 123, so that upper- and lowercase letters are adjacent. This is a clever scheme for many business applications. In systems that allow index expressions, the alphabetic adjacency can be achieved by forcing the index field to uppercase in the index expression. With DataFlex an index order would be "Smith, Smythe, smith" while in other systems the index order would be "Smith, smith, Smythe". These are distinctions of which the applications developer must be aware.

Because of the numeric relationships that DataFlex holds between files and between a file and its indexes, a developer must pay careful attention to a file definition that changes. Inserting a field can be disastrous unless the file interrelationships are manually maintained. In the sample application developed by *PC Tech Journal* for testing DBMS products (see "Evaluating Data Managers as Development Tools," Julie Anderson, this issue, p. 46), File 2 (the Article file) has two fields, Field 6 (Author Last Name) and Field 7 (Author First Name), which relate respectively to File 1 (the Author file), Field 1 (Last Name), and Field 2 (First Name). Altering the Author file by inserting a single-character field (Author Middle Initial) as Field 2 causes all subsequent fields to be renumbered. If the Article file is not also maintained correctly in order to relate its Field 7 to File 1, Field 3, database integrity will be lost.

These cautions are equally in order for situations in which REINDEX commands are built into the application. REINDEX refers to fields by number and must be modified to reflect any changes in the field numbers. DataFlex reorders the index tags in the file definition when a field is inserted, and it notifies the user that the indexes need to be rebuilt. Nonetheless, a developer must be wary of the effects of any changes to a database.

To add a new field, several other maintenance chores must be performed manually by the developer. First, a copy of the old file definition needs to be saved. Second, a QUERY must be performed to select all records and fields so that the database can be copied to an ASCII text file. Then the structure can be changed and the old data erased from the file. Finally, a DataFlex application named READ is used to create a map of the ASCII file into the new structure. READ does not perform the actual work, but generates a DataFlex program that does the work. It must be compiled and then run to complete the file-structure change process.

DataFlex has no interactive command mode. For any use of a file (other than QUERY), a program must be written and compiled. No built-in commands exist to allow access to a file on a record-by-record basis. No global file operations can be used to copy or delete selected records. Managing files known to the configuration file (creating and deleting data files) must be accomplished through the FILEDEF program; otherwise, database network integrity is sure to be lost.

DATABASE MAINTENANCE

DataFlex includes a proprietary programming language that is robust and comprehensive. Central to its use are

two commands: ENTER and REPORT. The ENTER command performs full-screen database maintenance complete with field-by-field editing with options for requiring an Entry, numeric range checking, and table look-up for validity checking. Most database maintenance tasks can be performed by using the standard ENTER program structure. Those that cannot either can use a substructure, the ENTER GROUP, or they can be programmed.

Simple ENTER structures, along with the few other program statements that are required to complete a DataFlex configuration (the name for a compiled program), can be generated by the AUTODEF utility program. AUTODEF is refreshingly simple to use. First the developer creates a screen form that describes the file to be created and maintained. A suitable form for the Author database in the sample application is shown in the top half of listing 1. Data fields and ASCII fields are indicated with underscores. Numeric fields are represented by a decimal point placed at the appropriate position. Dates are shown by —/ —/ —.

Next the AUTODEF program is run. It reads the screen-form text file and notes each data field. AUTODEF prompts for a name for each field and creates the file definition. It then creates any index files specified and initializes them. It also generates a file maintenance program that will use the ENTER command. The 26-line configuration source code generated by AUTODEF is shown in the bottom half of listing 1. This program is the minimum that is required to maintain a DataFlex database.

The DataFlex program created by AUTODEF allows the Author database to be maintained completely. Records can be added and deleted; they can also be found via any index to be viewed or updated. Each of these functions are accomplished by DataFlex in conjunction with the screen form defined, the simple program, and the 16 *flex keys.*

These keys are predefined for use with the ENTER program structure. DataFlex allows flex keys to be programmed for custom use. In essence, a user-defined subroutine is called when these keys are pressed. Much of the end-user experience with DataFlex is through the flex keys.

The AUTHOR configuration (program) is started from the DataFlex menu (a CHAIN AUTHOR internal command) or from DOS (a FLEX AUTHOR command), and the end user is presented with the previously defined screen form. At this point, the user has several options. Keying data to fill in the form defines a new record for the file. Pressing the HELP flex key (F1) will cause any developer-defined help screen associated with the data entry screen to be displayed. Because AUTODEF does not create such a screen, F1 has no effect. To find an existing record, the user can move the cursor to an indexed field by using the up and down arrow keys, then fill in the field completely or partially to match an existing record. After the field is entered, the user presses the SUPER FIND flex key (F3). DataFlex responds by finding the matching record to the entry or, if no match occurs, by finding the first record that occurs after the record, in index order. The fourth option is to press the ESCAPE flex key, which causes DataFlex to close the files and exit to the main menu.

When all the screen fields have been filled, DataFlex saves the record and displays the data entry screen with blank fields. The same effect is achieved in response to the user pressing the SAVE flex key (F10) from any field within the screen. And finally, either the PREVIOUS RECORD flex key (PgUp) or the NEXT RECORD flex key (PgDn) can be pressed to file the record and retrieve the preceding or succeeding record in index order. This last operation is a little tricky, because DataFlex chooses the previous/next record via the index for the field in which the cursor is currently located.

The 16 flex keys are predefined for use with the ENTER program structure. DataFlex allows them to be programmed for custom use.

To move to the next data maintenance action, the user presses the SAVE flex key (F10), clears the fields back to underscores with the CLEAR flex key (F9), and makes entries to find the next record to maintain. To delete a record, simply find the record by any appropriate means, then press the DELETE RECORD flex key (F6). To print a screen form on the printer, press the PRINT SCREEN flex key (F4).

An unusual CALC key (F5) allows any numeric expression to be entered on the bottom line of the screen. The expression can contain +, −, /, and * arithmetic operators as well as parentheses. The result of the expression's evaluation is then automatically placed in the current screen field.

Automatic record look-up and data validation are missing from the AUTODEF-generated program. Also missing is DataFlex's ability to display fields from related databases on the screen concurrently.

The file maintenance program that AUTODEF generated uses the ENTER program structure described earlier. Within that structure (see listing 1) are several ENTRY statements, such as ENTRY AUTHOR.LAST_NAME (meaning the AUTHOR database, the LAST_NAME field). Each entry statement maps in order, a database field to the screen field (in this case, to the right of the text Author:). Processing fields is usually easiest from left to right, top to bottom as they appear on the screen, and that is accomplished with the AUTOPAGE statement. Screen fields can also be referred to by number within DataFlex statements using the following form: screen_name.field_number (for example, MYSCREEN.1).

Each ENTRY statement can have options appended to it that tailor DataFlex's response to each field's contents. The 15 entry options vary in effect from {AUTOFIND}, which causes DataFlex automatically to find a record matching an entry into a screen field, to {RANGE = 1,10}, which performs a numeric range check. The ENTRY options can validate entries, including checking valid string responses; force uppercase text; skip entry of data in fields where values are filled in from a record in a related file; display fields from related records without allowing them to be edited; and specify the number of decimal points in a number. Many data validity checks or multifile operations can be performed automatically by DataFlex from within the ENTER program structure.

For those situations that require validity checking or other processing not provided by the ENTRY statement, DataFlex can turn processing over to explicit, detailed DataFlex statements that the user provides. When a flex key is pressed to terminate screen processing, DataFlex automatically calls any user-defined procedures that have been associated with that screen.

Within each of these procedures, an ERROR can be declared. Errors force DataFlex to discontinue further processing of the command, to display an error message at the bottom of the screen,

AZTEC

Manx Aztec C86 is the best C for MS-DOS and you can prove it yourself!

"A compiler that has many strengths ... quite valuable for serious work"
Computer Language review, February 1985

Manx Aztec C86 - The C For MS-DOS

Manx Aztec C86 is clearly the best C software development system for MS-DOS. Aztec C86 is the only C compiler for MS-DOS that provides the level of performance, features, documentation, and support required for serious, professional software development. You can prove it yourself. All you have to do is order Aztec C86 from Manx, evaluate it, and, if you like it, keep it. If you don't like it, send it back within 30 days and we'll cancel your order.

If you keep your Manx Aztec C86, as 99% do, you'll be in with the best company.

Manx Aztec C86 Features:

Optimized C compiler: Unsurpassed for code quality and speed. Optionally generates 80186 and 80286 code. Full K & R.

Symbolic Debugger: Execution trace, break points, display data in floating point, integer, character, or hex format. Evaluate expressions. Detect illegal memory stores, modify memory/registers, disassemble code.

Manx AS86 Macro Assembler: Supports macroes, 8086, 80186, and 80286 instructions in Intel format. Fast execution.

LN86 Overlay Linker: Links small, large, and mixed memory model routines, supports overlays, and options for producing ROM based code.

Librarian: Build and modify personal or system run time libraries.

8087/80287 Sensing Library: One library simulates floating point, another assumes the presence of an 8086 or 80287 math chip, the third senses the existence of a math chip, and if it finds one it uses it.

Profiler: Provides a run time analysis of your code to pinpoint code segments to optimize.

UNIX Library: Compatible with UNIX C. Fast I/O. Terminal I/O can be buffered or unbuffered.

DOS Library: Time and date functions, program forking (exec), program chaining, directory commands, I/O port support, sysint support, BIOS functions, and BDOS functions.

Screen & Graphics Library: Screen and cursor functions. Fast routines for drawing lines, circles, elipses, points, and setting colors.

CP/M-86 Library (-c): Produce programs for CP/M-86.

Large Memory Model: Manx Aztec C86 supports programs and data of any size. Global data has a max size of 64k.

Intel Object Option: Interface to software that requires Intel object format, such as PLINK86.

Z (vi) Source Editor (-c): Fast, powerful editor, Macro capabilities, undo, ctags, buffers for commands and data, and all the bells and whistles that make vi fanatics fanatical.

ROM Support Package (-c): Startup routine, linker options for separate placement of code and data, special utilities like the Intel HEX Utility, documentation, and library source.

Library Source Code (-c): UNIX, screen, graphics, and math function libraries.

Mixed Memory Models (-c): Mix large code and small data, small code and large data, or mix within type.

UniTools (-c): The UNIX utilities make, diff, and grep.

One year of updates (-c): As new versions are released, updates are automatically sent.

Technical Support: Manx has a full time staff to provide support via telephone & bulletin board.

Items marked -c are special features of the Aztec C86-c system.

Manx Aztec C86 is available in four configurations: Manx Aztec C86-c, Manx Aztec C86-d, Manx Aztec C86-p, and Manx Aztec C86-a. The -p and -a systems are not intended for commercial work and do not incorporate the same compilers as the -c and -d systems. All systems are upgradable.

Aztec C86-c Commercial System	**$499**
Aztec C86-d Developer's System	**$299**
Aztec C86-p Personal System	**$199**
Aztec C86-a Apprentice System	**$ 49**

Manx Cross Development Systems

Manx Aztec C compilers are available as native or as cross development systems for PC-DOS, MS-DOS, Macintosh, CP/M-86, CP/M-80, TRSDOS, Apple II, and Commodore 64/128.

Cross development involves two computer systems: the development system (HOST) and the execution system (TARGET). This method is useful when the TARGET machine is slower or more limited than the HOST.

HOSTS: VAX UNIX ($3000), PDP-11 UNIX ($2000), MS-DOS ($750), CP/M ($750), Macintosh ($750), CP/M-68k ($750), XENIX ($750).

TARGETS: MS-DOS, CP/M-86, Macintosh, CP/M-68k, CP/M-80, TRS-80 3 & 4, Apple II, Commodore C64, 8086/80x85 ROM, 68xxx ROM, 8080/8085/Z80 ROM, 65xx ROM.

Additional TARGETS are $300 to $500 (non VAX) or $1000 (VAX). Call for information, on cross development to the 68000, 65816, Amiga, C128, CP/M-68K, VRTX, and others.

How To Become a Manx Aztec C User

Call 1-800-221-0440 or 1-800-832-9273 (800-TEC WARE). In NJ or outside the USA call 201-530-7997. Orders can also be telexed to 4995812.

Payment can be by check, COD, American Express, VISA, Master Card, or Net 30 to qualified customers.

Orders can also be mailed to Manx Software Systems, Box 55, Shrewsbury, NJ 07701.

For More Information: call 1-800-221-0440, or 201-530-7997, or write to Manx Software Systems.

Manx maintains a large professional staff to service and support Manx users. You will get fast delivery and great service dealing directly with Manx.

Support Software for Manx Aztec C86

C-tree $395: B-tree database system. Easy to use. Available for Aztec C for MS-DOS, Macintosh, CP/M-86, CP/M-80, and others. Includes source.

PHACT $250: Powerful database system. Available for Manx Aztec C compilers for MS-DOS, CP/M-86, CP/M-80, and Macintosh.

PANEL $295: Create screens via simple, powerful editing commands. Select colors, edit fields. Directly input data to a multi-keyed file utility included with the system.

SunScreen $99: Create and modify formatted screens easily. Validate fields, select colors, create screens for both the color and monochrome cards. With library source SunScreen is $199.

WindScreen $149: Combines SunScreen with a powerful window utility.

Windows for C $195: Versatile window utility that supports IBM PC compatible and some non-compatible environments.

AMBER Windows $99: Powerful, low priced window package.

HALO $250: The ultimate C graphics package. It supports viewports, shapes, and multiple graphics cards. A less expensive version is available for just the PC mono and color cards.

FirsTime $295: Syntax checking while you edit greatly shortens compile time.

Pre-C $395: Powerful Lint-like utility locates structural and usage errors. Easily checks multiple files for bad parameter declarations and other interface errors. Lint users will find the user interface a dream come true.

PC-LINT $98: Lint-like utility that supports large memory models, has clear error messages, and executes quickly, has lots of options and features that you wouldn't expect at this low price.

Greenleaf Functions $185: Source for over 200 C and assembler functions. They are great, they work, they are used extensively, and are economically priced. Clear documentation and easy to use interface round out an impressive package.

C Utility Library $185: C and assembler source for screens, windows, color graphics, asynch communications, and more. The color graphics and speed of this package are impressive.

Plink-86 $395: MS-DOS linkage editor for producing and maintaining overlayed programs. It works with Aztec C86 in Intel object format mode.

30 Day Guaranty:

Any Manx Aztec development system can be returned within 30 days for a refund if it fails to meet your needs. Restrictions are that the original purchase must be directly from Manx, shipped within the USA, and the package must be in new condition. Returned items must be received by Manx within 30 days. A restocking fee may be required.

Discounts:

There are special discounts available to professors, students, and consultants. A discount is also available on a "trade in" basis for users of competing C systems.

Manx Aztec C Distribution:

In the USA, Manx Software Systems is the exclusive distributor of Aztec C. Telephone or mail order sales other than through Manx are unauthorized.

MANX

To order or for information call:

800-221-0440

CIRCLE NO. 208 ON READER SERVICE CARD

and to return to the screen for further editing. Thus, DataFlex will automatically display an error "INVALID ENTRY FOR THIS WINDOW #15" when text entered in a field does not pass the programmed validity check. But the applications developer can choose not to use DataFlex's validity check and instead perform his own check in the ENTRY.EDIT: subroutine. If within that subroutine the code determines that an entry is invalid, a custom error message and number can be declared, such as "ERROR 101: Not a valid state—Press F1 to display valid state abbreviations."

For those developers who like the standard ENTER option processing, but dislike the wording of its error messages, yet another alternative is available. DataFlex is provided with a complete error message file maintenance program (a DataFlex configuration), so that error messages can be customized.

By adding ENTER options, a help screen, and a custom ENTRY.SAVE: subroutine to the AUTODEF-generated program, a developer can create a comprehensive database maintenance routine. The modified program for the Author database is shown in listing 2 (the screen portion is unchanged from listing 1), and the file definition created by AUTODEF is shown in listing 3.

A multifile maintenance program can be written starting from an AUTODEF-generated program and then adding multifile statements. Each file to be used must be OPENed and named in the ENTER statement. Within the screen form, each field's contents are designated by the FILENAME.FIELDNAME syntax, so fields from different files can be freely mixed on one screen form. Through judicious use of AUTODEF and by tailoring the AUTODEF-generated programs to suit particular situations, a programmer can develop entire multifile applications without learning the fine points of the DataFlex language.

THE QUERY PROGRAM

DataFlex includes a program named QUERY that allows the novice end user to create ad hoc multifile reports. All the choices a user has at each step of the query formulation are displayed, including file names, field names, and selection operators.

QUERY makes extensive use of cursor pointing—the technique of highlighting one selection on a screen. The user pushes the arrow keys to move the highlight to the desired entry, then presses Return to confirm a selection.

Seven record-selection operators are available: equal to, not equal to, greater than, greater than or equal to, less than, less than or equal to, and including. Substrings can be designated with the including operator. Multiple selection criteria form a logical conjunction (AND), and QUERY supports multifile selection criteria for related files. Output can be sorted in any of the index orders for the primary database.

Fields from the primary database and any related database can be selected in any order. Numeric type fields can be totaled, and line-wrap can be specified to occur at selected fields rather than at QUERY's default column location. QUERY produces a columnar report of records that match the selection criteria. The report may be displayed, written to a file, or printed.

Printed reports can be formatted for page breaks or mailing labels (only one-up). Output disk files can be created as report images suitable for later printing, mailing labels, a fixed-length ASCII file (each record terminated with CR/LF), or a comma-delimited ASCII file (readable by BASIC, dBASE II, and similar programs). QUERY can also generate the source code for a DataFlex program to recreate the query at any time from within DataFlex proper. QUERY is thus a report program generator as well as an ad hoc inquiry program. This latter point is very important, because it parallels AUTODEF in functionality and allows developers to create report programs without any knowledge of the DataFlex language itself.

DataFlex includes a program named QUERY that lets the novice user create ad hoc multifile reports.

QUERY-generated source code for the sample application is shown in listing 4. The generated code uses a DataFlex nonprocedural program structure called REPORT, which is similar to the ENTER structure that is used for database maintenance. Within the REPORT structure are the following forms layouts: the report heading, /HEADER, to be generated at the top of each page; the detail lines, /BODY, to be generated for each record selected from the primary database; and the report summary totals, /TOTAL, to be printed at the end of the report. The record selection criteria prompts can be found in the /SELECTION section.

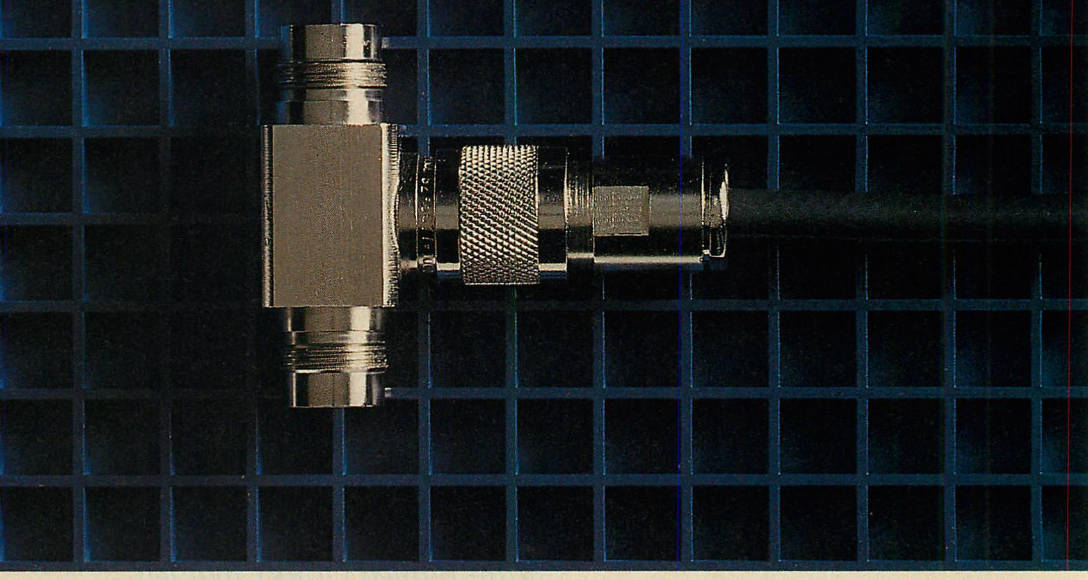
Following the forms area within the report program is the REPORT structure itself—in this case the line REPORT ARTICLE BY INDEX.1. The SECTION HEADER, SECTION BODY, and SECTION TOTAL portions of the report structure include the detailed processing to take place within each section of the report as it is generated. This report is directed to the printer. To direct the report to the screen instead, the statement OUTFILE should be changed to OUTFILE CON:. The addition of about 15 DataFlex statements at the beginning changes the program to prompt the operator for the report destination at runtime.

THE DATAFLEX ENVIRONMENT

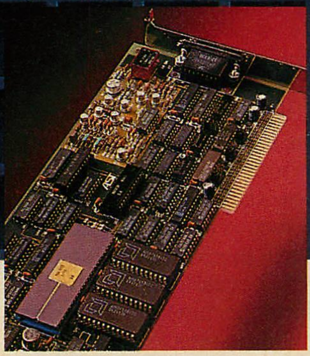
The central activity of a DataFlex application can be thought of as transporting data between various data sites—from a data entry screen (window) field to a database element, from a window field to a memory variable, from a memory variable to a database element, and so on. These data movements establish the character of the DataFlex language.

DataFlex is rigorously modular in its approach to building a complete applications system; this is best seen in its approach to menus for the application's end user. When invoked, DataFlex runs MENU.FLX. This menu program has the job of displaying menus, allowing items from a menu to be selected, checking for any required password, and initiating the desired function. The menu can prompt with two questions to complete the information needed to perform a function, and one menu can lead to yet another. Each menu has a "return-to" menu to which it passes control when Esc is pressed. Many DOS system functions can be performed from within menus, such as displaying a directory list, or copying a file. Unfortunately, the menus, and indeed DataFlex itself, do not support DOS paths for file names, either implied or explicit. DataFlex can operate only on the current directory for each available drive. (See "File-search Help for PC-DOS," Arthur Gleckler, *PC Tech Journal*, April 1985, p. 138, for ways around this problem.)

In the database maintenance program example above, the DataFlex program to maintain the Author file is stored in the file called AUTHOR.FLX. The menu runs this program with the command CHAIN AUTHOR. When the author program is terminated by the end user pressing Esc, the DataFlex runtime system automatically chains



Direct PC to System 34/36/38 Connections Using Standard Twinax Cable



It's Easy With AST-5251/11:™ Fast Hookup, File Transfer And Extra Capabilities

Standard Twinax Cable Support Makes Installation A Snap. AST-5251/11 and twinax connection means easy, cost effective direct cable attachment of your PC to your IBM® System 34/36/38—no expensive communications adapter necessary. Just plug our AST-5251/11 emulation board into your PC/AT/XT, portable or compatible and connect the twinax cable. With this complete hardware/software solution that's all it takes to use your PC just like a standard IBM 5251/11 terminal.

File Transfer Is Only The Beginning. Use your PC as always, and emulate a Model 11 terminal—with powerful extra capabilities—to access your host minicomputer. Then transfer files bi-directionally between your host and PC, in a ready-to-use format.

Use Your Favorite PC Software. AST's file transfer support not only eliminates the burden and the errors of re-keying, it adds the ability to manipulate transferred data using your favorite PC software packages, like Lotus 1-2-3™ and WordStar.® And our Hot-Key support lets you conduct concurrent DOS and host sessions, while our applications program interface (API) makes integrating PC and host applications simple.

Even Greater Efficiency. AST-5251/11 also provides host-addressable 5256 printer emulation for your PC printer in a background mode.

See your dealer today, or call our Customer Information Center (714) 863-1333 for more information. AST Research, Inc., 2121 Alton Avenue, Irvine, CA 92714 TWX: 753699ASTR UR.

KEY FEATURES

- *Plugs into PC/XT/AT and compatibles; connects to System 34/36/38 via twinax cable and supports cable thru.*
- *Emulates 5251-11, 5291, or 5292 -1 display terminal.*
- *Provides host-addressable 5256 printer support (background mode) on PC-attached printer.*
- *Selectable DMA and I/O interrupt channels.*
- *Application program interface and bidirectional file transfer included.*
- *Complete package includes hardware, software, twinax cable assembly, and comprehensive user manuals.*
- *Supports fixed or hard disk.*
- *Hot-Key assist for concurrent host and PC sessions.*

AST
RESEARCH INC.

AST-5251/11 trademark of AST Research, Inc. IBM registered trademark of International Business Machines Corp. WordStar registered trademark of MicroPro International Corp. Lotus and 1-2-3 trademarks of Lotus Development Corp.

CIRCLE NO. 103 ON READER SERVICE CARD

back to the menu from which it was started. Returning to a DataFlex menu is a persistent practice throughout the entire DataFlex system.

AUTODEF, for example, will chain to the compiler to compile a program it generates, then the compiler will chain to the DataFlex main menu when it is done. In fact, the menu system as delivered with DataFlex comprises a complete program development system. From it can be run all the DataFlex utilities as well as non-DataFlex programs, such as a text editor.

The net result is that DataFlex automatically builds a system from interrelated programs through its menu system. DataFlex programs can actually be run from DOS (in this example, the command is FLEX AUTHOR), but DataFlex will return to the menu on termination of the author program. The only way to bypass this is to include within the author program a user-defined exit subroutine that uses the command SYSTEM instead of RETURN.

But why fight it? Each time a program is created, place an entry for it within a DataFlex menu so it can be tested and used. Such menu entries are created using the MENUDEF utility. In the spirit of DataFlex's "it's your system" approach, MENUDEF maintains a DataFlex file containing the menu information. MENU.FLX, the menu program, is also supplied in source form.

THE APPLICATIONS LANGUAGE

The explanations given above of the AUTODEF-generated database maintenance programs and the QUERY-generated report programs indicate that much of an application can be created without programming in the DataFlex language. In fact, the power of DataFlex, like that of most DBMSs, cannot be realized without a complete command of the language itself.

The DataFlex language is unique. On the one hand are comprehensive structures, such as ENTER and REPORT, as described above. On the other hand are more typical and mundane statements, such as INKEY key_answer, which waits for a single key to be pressed on the keyboard, and SHOW, which displays a message on the screen. In between are the assignment statements, I/O statements, repetition structures, branching statements, and modular program statements of most high-level languages. Some DataFlex statements resemble BASIC; others are similar to Pascal or COBOL.

A DataFlex program's source code must be compiled before it can be used, and it cannot be used without the DataFlex runtime system. Actually, the code is semi-compiled into data arguments and functions for the runtime system to perform. Unlike stand-alone compiled languages, DataFlex compiled code can be used immediately and need not be linked to a runtime library.

The DataFlex language is rich and divides into 15 command groups. Thus, it has groups of statements to support movement of data between the screen, variables, and files explicitly and implicitly with the ENTER and REPORT structures; it has statements to manage the window environment, called the forms group; and it has the database statement group. Other groups are for program control, structured control, console I/O, sequential I/O, conditional execution, and system commands. Finally, there are the memory variable definition group, string operations, the key-stroke command (flex key) group, and the multiuser commands.

The power of DataFlex cannot be realized without a complete command of the language itself.

Program variables can be numeric, string, integer, date, or logical type and must be declared by type before they can be used. Multiple declarations on one line are allowed. Therefore,

```
NUMBER past_due_30 past_due_60
```

declares two numeric variables. Variable names can be up to 80 characters long and can use underscores (but not spaces) to increase readability.

Each statement in the DataFlex program source has the form:

```
<verb> <object> . . . <object>
```

Program statements can be up to 255 characters long, and logical lines can span several physical lines if a semicolon is placed at the end of each physical line. Labels begin a line, end with a colon, and can be up to 80 characters long. Within the line, white space (blanks used for indentation or program structure emphasis) and case are ignored by the compiler. Each source line can be commented; comments start with a double-slash. For example:

```
MyLabel: MOVE (2+2) TO four   // silly
  example
```

DataFlex cannot tolerate complexity in its program statements. Calculations, for example, must be explicit stand-alone assignment statements of the form CALCULATE <expression> TO <argument>. Arguments as they appear in other commands, in particular in the I/O commands and the destination argument in the MOVE command, cannot be expressions themselves.

A similar intolerance for complexity underlies how the logical variables, called *indicators*, are used. The DataFlex IF statement supports eight different comparison modes, but does not support ANDing or ORing operations together into one statement. These tasks are left to the more sophisticated INDICATE statement, which evaluates a statement and sets an indicator to True or False. It is the INDICATE statement that allows AND and OR statements to be strung together. For example, the following statement sets the variable *eligible* True if a person is male, single, of age, and either rich or handsome.

```
INDICATE eligible GROUP ALL [male single
  of_age]; AND ANY [rich handsome]
```

The indicators used, *male*, *single*, and *of age,* are limited to three within a group, and only two groups can be present in a command. DataFlex is not limited in what it can logically do by these restrictions, but the user may be surprised at the number of successive commands required in some situations.

One curious aspect of the language is its lack of mathematical functions. Generally, languages provide functions to help do complex mathematical calculations, string operations, or data type conversions. Missing from DataFlex are functions such as SIN, COS, and SQRT. Also missing are rounding functions, exponentials, and logarithms.

String operations are present in the language as single-statement assignments. Although these may be cumbersome to use, the repertoire is comprehensive—similar to the problems mentioned earlier with complex logic tests. Included are statements to extract characters from the left, right, and middle of a string, to determine the position of one string within another, to determine the length of a string, to pad or trim trailing blanks, to insert or replace one string within another, to append strings together, and to force a string to uppercase. No statement exists to convert a string to lowercase. Conversions between character and numeric data types are supported by the language.

DataFlex supports date variables in databases, windows, and memory vari-

SIM3278/PC

A MICRO-MAINFRAME CONNECTION IN A CLASS OF ITS OWN

Data Decisions Gives SIM3278/PC High Marks

Data Decisions reviewed and evaluated SIM3278/PC, giving it high marks for* ***User Interface, Ease of Use and Systems Interface.* *Their conclusion: "Simware has demonstrated an understanding of the user's need to maximize productivity gains possible with computers."*

Using inexpensive asynchronous or X.25 network communications, SIM/PC provides you with advanced features such as error free file transfer, printer support and an intelligent command processor. "Unlike many communications packages that present primarily a menu-driven user interface, SIM/PC's command language is a very powerful enhancement over simple 3270 emulation."

SIM/PC eliminates the need for expensive add-on boards and gives you the most flexible micro-to-mainframe link available.

A corporate license entitles you to make multiple copies of SIM/PC without paying any additional per copy price every time another PC is added to your network.

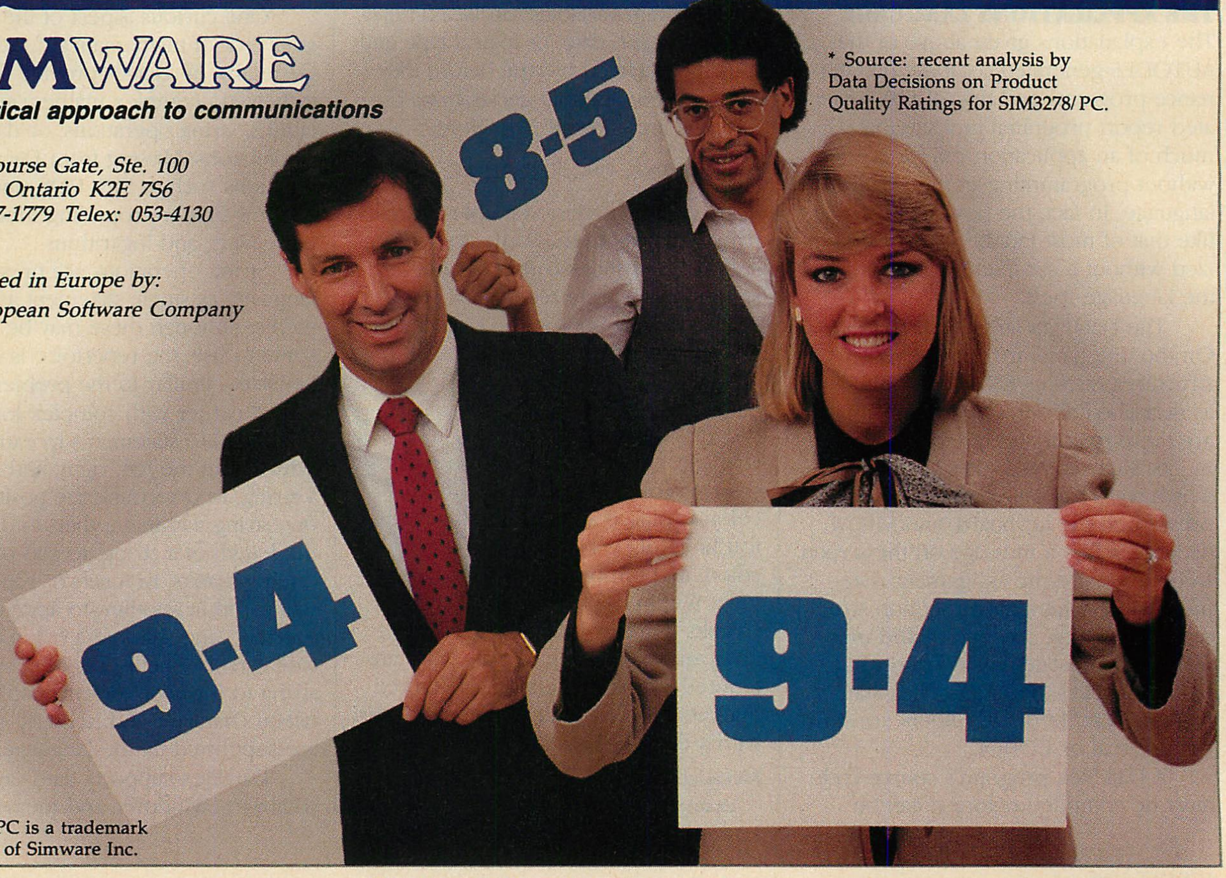
CALL TODAY FOR A FREE EVALUATION SYSTEM AND JUDGE FOR YOURSELF.

SIMWARE
a practical approach to communications

14 Concourse Gate, Ste. 100
Nepean, Ontario K2E 7S6
(613) 727-1779 Telex: 053-4130

Distributed in Europe by:
The European Software Company

* Source: recent analysis by Data Decisions on Product Quality Ratings for SIM3278/PC.



SIM3278/PC is a trademark (pending) of Simware Inc.

CIRCLE NO. 188 ON READER SERVICE CARD

ables; it also supports expressions mixing numbers and dates—for example, 8/19/85 + 30. Internally, DataFlex can accommodate the years from 1 A.D. to 2500 A.D. It also can provide a statement to retrieve the system time and date. However, it contains no functions for determining the day of week or the calendar month or for extracting the year, day, or month portion of a date variable. In addition, it lacks a provision for converting between character strings and dates.

DataFlex can be used for data input and output outside of the database structures with I/O commands for numeric and character input, line input, line display, clearing the screen, and direct cursor addressing. The latter includes a complete set of sequential I/O statements similar to those of BASIC, so ASCII files can be read and written from a DataFlex application (these can also be sent to the serial port for use with a modem, although the language lacks complete modem control).

Finally, DataFlex cannot test for the existence of a file. This is important for users who place reports on disk for future reference and need assurance that they will not mistakenly overwrite an existing report file.

DataFlex is not a strictly structured language and freely supports GOTO <label> and RETURN <label> statements. Instead of a case structure, DataFlex uses a BASIC-like ON <integer arg> GOSUB <label> <label>... <label> statement and an ON <integer arg> GOTO <label> <label>... <label> statement. Program repetition statements include BEGIN/END (generally used with preceding indicators), REPEAT/UNTIL and REPEAT/LOOP structures, a WHILE/END structure, and a FOR/FROM/TO/LOOP structure.

Modularity within a DataFlex application is provided primarily by chaining from a menu to an independent program, which then returns to the menu. Communication between modules is available through a common area containing as many as 40 integer variables, through single-record system files created by saving the current memory variables to a file, and through command line parameters. Command lines provide a convenient way for one program to send runtime information to another that it starts. This is particularly effective when used with the menu system. For example, a report can place its output on the screen or printer, based on its starting command. Within the menu system different entries can distinguish the output destinations.

Within a single program, subroutine structures are available through the GOSUB <label> and RETURN <label> statements, up to a nesting depth of 20 procedures. Within a module, all variables are global.

One unusual aspect of the DataFlex language is its predefined indicators (the True/False logical variables). DataFlex maintains 38 predefined indicators that can be used to test for errors, check which flex key terminated a window, check end-of-file conditions, and determine various REPORT structure conditions, such as which breakpoint triggered a subtotal.

DataFlex is generous in many of its language capacities. A DataFlex program can have up to 5,000 lines, each of which can be 255 characters long.

DataFlex also includes 17 predefined system variables that can be used to control movement between windows, to retrieve the page number of a report, and to determine several aspects of the file definition. In combinations, the predefined indicators and variables allow for processing somewhat akin to software interrupt routines.

DataFlex is generous in many of its language capacities. A DataFlex program can have up to 5,000 lines, each of which can be 255 characters long. As many as 10,000 variables and 89 indicators can be defined, in addition to the 38 predefined indicators. Curiously, only 40 integer variables are allowed. This limitation is required, in part, because when one configuration chains to another, the values and names of the integer variables are preserved.

THE DEVELOPMENT PROCESS

A DataFlex source program is transformed to work with the DataFlex runtime system in two steps. The first step preprocesses the source program to expand the more complex DataFlex statements into simpler statements that the runtime system can handle. The second step creates the actual functions and arguments for use with the runtime system. Understanding this process and its purpose is not essential for its use, but does lend insight into both the limitations and potential of DataFlex.

Many of the DataFlex statements are macro commands—short-cuts for referring to longer sequences of statements. Expanding the macros is the primary purpose of the preprocessor. Part of the expansion is a substitution of file element references for the <filename>.<fieldname> syntax used in the language. AUTHOR.FIRST_NAME may be replaced by |FS1,2 denoting the first file, second field in the overall configuration. *F* designates the reference to a database field, and *S* means the field is a string type. The OPEN AUTHOR statement (listing 1) within the language causes the compiler to generate functions to open the file, but also causes it to read in the file's definition from the AUTHOR.FD file. Ultimately, each DataFlex statement takes on the form

```
<line number> <ind1> <ind2> <ind3>
  <command> <arg1> <arg2>
```

The expression *ind1* refers to the first indicator for execution of the command, and the syntax shows there can be up to three. In the statement, *command* refers to the internal command number, essentially a function for the runtime system to perform; *arg1* refers to the first of two possible arguments on which the function operates.

Insight into the DataFlex process can be gained by examining the possible composition of the arguments. FS, in the example above, codifies the class and type of an argument as field and string, respectively. Classes designate constants, memory variables, window contents, a database field, or a group structure (ENTER uses a group structure). Types designate strings, numbers, dates, integers, expressions, labels, group options, and group indicators.

The line number in the DataFlex internal statement syntax refers to the generated command sequence number, not necessarily the source code line number. This is the line number shown by compiler error messages, by runtime error messages, and by debug tracing. The compiler shows the first internal line number generated for a source line in the listing.

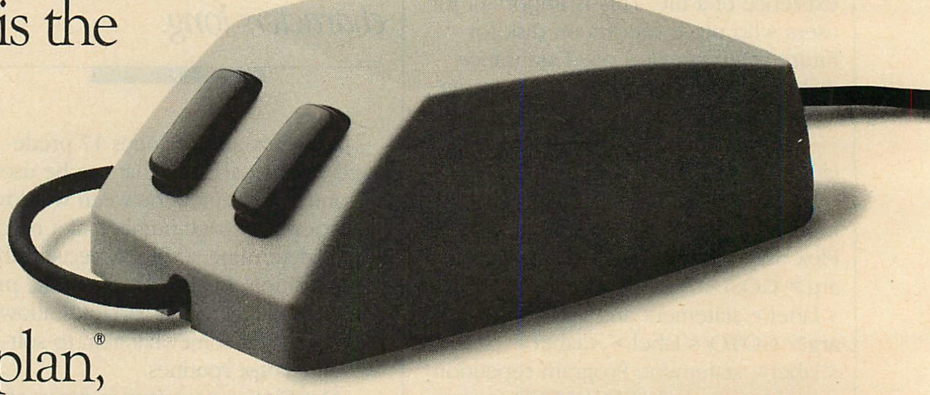
A number of command line switches affect the compiler's actions. These determine whether it will pause when an error is encountered, whether the listing is placed on the screen, printer, or disk file, and the destination drive letter. The source code can include preprocessor directives for advanced use. An important one is the #INCLUDE directive, which adds the

Buy Mouse, Paint Free.

Now when you buy the Microsoft® Mouse, you get a dazzling new color paint program, PC Paintbrush™ free.

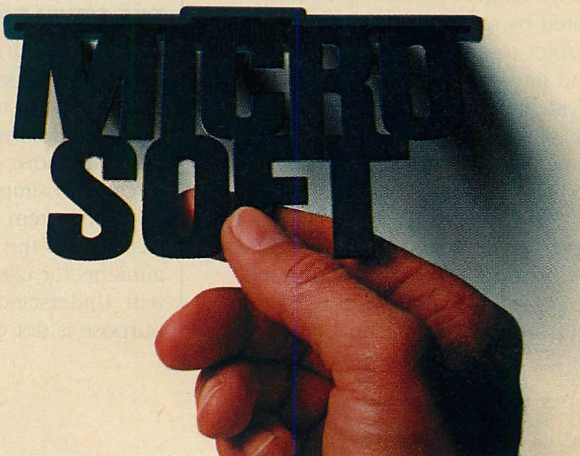
PC Paintbrush supports a host of graphics cards (including the new IBM® EGA and Hercules™ Graphics Card) and over 30 printers and plotters.

Microsoft Mouse is the high performance mouse for the IBM Personal Computers. It puts the power to run Microsoft Multiplan,® Lotus® 1-2-3,® Microsoft Word and other popular programs in the palm of your hand. It's available in serial and bus versions.



MICROSOFT®
The High Performance Software™

So why monkey around with any other mouse. Call (800) 426-9400 for your nearest Microsoft dealer. In Washington State, Alaska, Hawaii and Canada, call (206) 828-8088.



Microsoft and Multiplan are registered trademarks and The High Performance Software is a trademark of Microsoft Corporation. Lotus and 1-2-3 are registered trademarks of Lotus Development Corporation. IBM is a registered trademark of International Business Machines Corporation. Hercules is a trademark of Hercules Computer Technology. PC Paintbrush is a trademark of ZSoft Corporation.

contents of an ASCII source file to the source file being compiled. The include facility allows developer-designed subroutines or other program fragments to be easily used across several configuration programs. Because included source code is not generally listed by the compiler, it is helpful to place "#REM Version of 11/20/84" or similar statements within include files; the compiler lists these during code generation. The compiler also responds to directives that set and test compiler variables, and it conditionally compiles source containing IF/ELSE/ENDIF structures.

Developers can even define their own DataFlex commands for the preprocessor to expand. In fact, DataFlex comes with the complete source code for the command expansions, so that they can be tailored if required. Beyond this, the developer can purchase an unlinked library form of DataFlex for $400, including some Pascal source.

Running the compiler takes about two minutes on a PC/XT for the short programs generated by AUTODEF and QUERY. More detailed programs take longer, of course. Considering that a DataFlex application consists of many individual programs tied together by the menu system, work continues at a fairly even pace throughout the development cycle. DataFlex resolves source code to database field references at compile time. Consequently, any change made to the structure of the databases—field location or field type—must be followed by recompilation of all the application's programs.

Program source code can be created using DataFlex's simple screen text editor or any editor that creates a standard ASCII text file. The DataFlex runtime system and all the utilities are compatible with RAM disks and with memory-resident programs. Most of the source code developed for this article was created using Borland International's SideKick, which permits program testing and source code editing to be coupled. In addition, SideKick allows programmers to create text files using the PC's extended graphics characters, which is quite helpful in designing screens.

Testing and correcting for errors within a DataFlex program can be aided with the DEBUG statement. DataFlex does not support breakpoints, but DEBUG turns on or off tracing of generated statements. When tracing is on, each statement's internal line number is displayed on the screen as it is executed; for example, [10] when line 10 is executed. This continues until another DEBUG statement is executed, at which point line tracing stops.

A few miscellaneous items complete the discussion of DataFlex's capabilities. The Intel 8087/80287 numeric processors are not supported by DataFlex. The language can be extended by using direct calls to DOS, but macro commands must be created to do so, and neither DOS nor assembly language routines can be called with the provided DataFlex commands. DataFlex's databases are never actually sorted, but are indexed instead. A sorted output file could be created using an index and the QUERY utility.

An unusual aspect of DataFlex as a DBMS is its BASIC-like sequential file processing command set. These commands can be used for a transaction log or for importing data from other programs.

REINDEX attempts to repair damaged database files, and it rebuilds indexes. REINDEX can be made to rebuild every index for every file in the configuration list with a single command, a convenience provided for applications installation on a customer's system.

DataFlex has no checkpoint/restart capabilities other than those the developer might program. Neither does it have built-in transaction logging, but it is designed to allow programmers to integrate transaction processing into an application. An unusual aspect of DataFlex as a DBMS is its BASIC-like sequential file processing command set. These commands can be used for a transaction log or for importing data from other programs.

Designed for use with multiuser systems, DataFlex provides record-level multiuser locking automatically within the ENTER program structure. Beyond that structure it provides complete applications-level control of a multiuser environment, including explicit record locking and unlocking, rereading data before processing, conditional execution of code dependent on whether or not a record buffer has changed, and output file despooling. Indeed, the applications design implied by DataFlex's fundamental concepts of moving data between data sites is ideal for multiuser designs that require data to be displayed and changed at a local level—with provision for the possibility of the actual data records being altered by another user during the local edit.

END-USER CONSIDERATIONS

In general, a developer delivers DataFlex applications to an end user, a person or organization who will use the application on a regular basis. The end user will not necessarily have any technical knowledge of the application—only how to use it.

One aspect that less experienced end users will appreciate is the menu system. Not only does the menu present a consistent view of the application to the end user, but other programs and system functions appear to the end user simply as other menu entries.

A second aspect of the end-user interface is the keyboard. DataFlex makes consistent and extensive use of the IBM PC keyboard. The Ins and Del keys trigger field editing functions that make sense and are consistent with other applications. Likewise, the arrow cursor keys can be used intuitively.

The ten function keys require a keyboard guide to be used effectively, and keytop stickers are supplied. More importantly, the ten function keys, combined with the PgUp and PgDn keys on the keypad, are essential to the effective use of a DataFlex application. Most daily users of a system have sufficient stake in its effectiveness to find the short training in using flex keys worthwhile. Ample help screens, which are displayed when the F1 key is pressed, may be sufficient for such installations. The benefits of the flex keys are consistency within the application and the programmed ability to support multiple user-directed program actions from any single point within a screen. DataFlex can vary in its response to flex keys, either through explicit key handling routines provided within an application or through the reassignment of such keys with the system installation program.

Effective use of the display screen is a third critical user interface. DataFlex addresses the PC video memory directly to achieve nearly instantaneous screen displays. The PC's extended character set is compatible with DataFlex, but the developer must use an editor that supports them. DataFlex supports a single screen attribute, high and low intensity, with its command lan-

guage. The single attribute is used throughout DataFlex to separate data from background information. The normal setting is for data to be shown in high intensity and background in low intensity. The installation program allows the low and high settings to be assigned to colors on a color monitor. Screen attributes cannot be set via ANSI escape sequences from within DataFlex, although sending Esc followed by a single "[" character sets the screen to reverse video. The standard command to set low intensity resets the screen.

DataFlex has no key floppy disk or other copy-protection scheme that the end user must handle, so its use on fixed-disk and network systems is without user inconvenience.

Ad hoc use of the system by those who are comfortable with file names, field names, and selection criteria is most effective with the QUERY facility. Similar considerations to the use of flex keys apply; the end user must be trained in order to use Query effectively. Provided that the developer has adequately met the application's requirements for integrated data maintenance and standard reports, QUERY can meet most occasional needs.

TABLE 1: *Benchmark Results*

BENCHMARK TASK	TIME (secs)
1 Add 900 records to an empty database	350
2 Index database on 2 fields (7 bytes)	84
3 Document and tally codes from one field	55
4 Mass change of one field (28 records of 900)	19
5 Extract selected records to create a text file	5

Although these numbers will become meaningful only when compared to the future results from other data managers, DataFlex's overall performance seems peppy.

THE SAMPLE APPLICATION

The FILEDEF program was used to create the data file structures, including the index definitions for the sample editorial inventory system used to test DataFlex. The READ configuration was used to read the ASCII data into the DataFlex files—a two-step process. READ generates the source code for a DataFlex program to perform the actual file loading; the program must be compiled before it is used. READ has an important option not mentioned in the DataFlex manual: if the incoming data match the destination database's file layout, READ generates the program with no further direction. Otherwise READ conducts a dialogue to determine the exact mapping of incoming data to destination fields on a field-by-field basis. The programs created by READ work perfectly for all three files.

Learning to use this portion of the DataFlex system seemed to take an appropriate amount of time after working through the tutorial and sample system provided with DataFlex. The sample is a fully working application, and the tutorial orients the learner to the DataFlex system through its use.

The FILEDEF program used to create the editorial inventory files has no on-line help, and several of its functions need to be tried experimentally to determine their effects. For example, unwanted databases cannot simply be deleted via DOS commands. Once deleted in this way, they cannot be omitted from DataFlex's file list. Instead, the FILEDEF "Set file inactive" function must be used to remove the file from the list. The same function has an option to delete the actual data file and its related index and definition files.

When the files themselves have been created, programs to maintain them have to be written. AUTODEF creates both a database definition and its maintenance program by processing a data entry screen. Unfortunately, it has no option *not* to create the data file. Thus, AUTODEF cannot be used to generate a program to maintain a pre-existing file. The user can create a new file with a different name and then modify the AUTODEF-generated program to maintain the pre-existing file. Or the file and field names of an existing AUTODEF-generated file maintenance program can be changed to suit the new files. Doing so forces the user to understand how DataFlex programs work—something not required for simple AUTODEF/QUERY use of the system.

Several report requirements, such as placing the time, date, and page number at the top of each page, must be added to the code that is generated by DataFlex.

The editorial inventory application requires a series of queries and reports to be created. The QUERY program is adequate to meet these requirements. Unlike AUTODEF and FILEDEF, QUERY has extensive context-sensitive help available. In many cases, it does not satisfy all the ad hoc inquiry requirements, but does provide adequate information. For example, the average fee paid to authors must be calculated by hand from the QUERY-generated number of authors and total fees paid.

QUERY has an option to generate a report that recreates the inquiry results, and that option is used to generate the basic report programs. Several report requirements, such as placing the time, date, and page number at the top of each page, must be added to the generated code. Adding a page number to a report is easy. A field is placed in the header screen image of the report (Page ___.), then the line PRINT PAGECOUNT TO header.1 is added to the header section of the report. The DataFlex REPORT program structure does the rest automatically. PAGECOUNT is one of DataFlex's predefined variables.

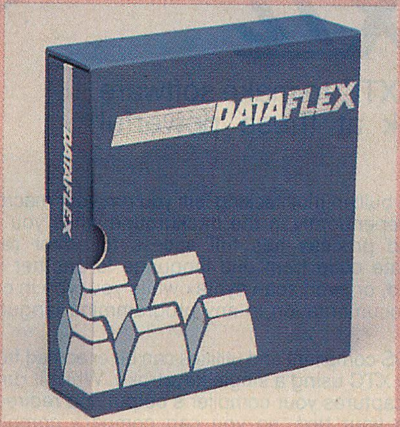
The third report requires subtotals at breakpoints on the author's name, so a /SUBTOTAL1 form and section are added. Subtotals are not calculated by QUERY; code must be added to the program that QUERY generates. DataFlex is told to generate subtotals with the addition of BREAK author_name to the REPORT ARTICLE BY INDEX.1 statement. This causes the /SUBTOTAL1 section to be processed when a new record does not match the previous record's author_ name. Obviously, the index chosen for the report plays a key role in what actually happens during the report generation.

Other requirements of the reports involve more extensive changes. The significance of the QUERY-generated report programs is that QUERY writes the program, then the developer adds enhancements. The starting point for the developer's changes is a correctly working program. Once a suitable method is found for adding routine enhancements, the changes to one report can be mimicked in later reports.

DATAFLEX OVERVIEW

DATAFLEX 2.4
Data Access Corporation, 8525 S.W. 129 Terrace, Miami, FL 33156. 305/238-0012.

Product type. Extended relational database management/applications-development system intended for a complete range of users. It includes utilities for novice users and an extensive command language for professional applications developers.
Software environment Runs under CP/M-80, CP/M-86, MP/M-86, Concurrent CP/M-86, generic MS-DOS, PC-DOS 1.x/2.x/3.x, TurboDOS 80/86, TeleVideo mmMOST, Action DPC/OS. Version running XENIX 3.1 is due to be released in July/August.
Hardware environment. IBM PC, XT, AT, compatibles, NCR DecisionMate, Molecular and other multiuser systems, TeleVideo 8-bit systems, Apple II with CP/M card, CP/M-86 generic systems, MP/M-86 generic systems, MS-DOS generic systems; 16-bit systems require 256KB of RAM and 600KB of disk storage; 8-bit systems require 52KB available memory. Hard-disk drive is optional. LANs supported: IBM PC Network, Orchid PCnet, Novell Netware, Digital Microsystems HiNet, Molecular N-Star, Corvus Omninet, 3com Etherseries, Micromation M/Net, TeleVideo Infoshare.
User Interface. Command interface: menu-driven with prompts; macros and procedures; program is command-language driven with English-like syntax; data definition is in English-like language with two levels of definition. Help facilities include context-sensitive on-line help and a quick reference card; a demo guide walks the user through keystroke by keystroke; the manual primarily used as reference but also has short tutorial.
File limitations. 255 characters per field; 255 fields per record; 65,000 characters per record; 64,000 records per file for 8-bit systems; 16.2 million records per file for 16-bit systems; memory limits number of open files. Four indexes per file for 8-bit systems; nine indexes per file for 16-bit systems; each index has up to six segments; maximum of 40 integer variables; numeric precision of numbers is four digits to right of decimal. Files cannot span multiple disk volumes; free-form data dictionary architecture.
File handling. The product can be pro-



grammed to merge two or more files and to split files; it has the capability to update multiple files simultaneously and to add fields without loss of data.
Input. Screen design: screen painting method of entry screen design; custom data-entry screens; multiple data-entry screens per file; entry screen can be greater than one screen in length; program allows derived fields as a result of calculations, from another file, from a list of acceptable responses; view-only fields; user-defined numeric fields; user-defined numeric formats; dollar formats and floating dollar signs; required fields. Data entry/edit: check for duplicate entry; range checking provided; facilities available for batch entry into file.
Query. Partial key search; comparison operators (>,<,>=,<=,=, string inclusion, string match); logical selection operators (AND, OR, and NOT); ascending sorts, sorts on up to six fields; 128 characters per index; query and ordering specifications can be saved for repetitive use; automatic updating of indexes.
Report generation. Label types of reports produced; "two or more across" labels can be programmed; report formats can be saved/modified/edited; reports can contain information from two or more files; summary reports can be produced; screen painting design method; nine levels of subtotals; control breaks. Reports contain calculated results; four-function math; parenthetical control of order of operations; averages can be programmed. Reports can include headings/footers/pagination; report output can be directed to screen, printer, choice of multiple printers by a temporary exit from the program, disk file, and import/export file format. Reports can include print enhancements such as bold and underscore; report definition provides opportunity to specify paper size and margins; user-programmable report specifications.
Security. Password-protected access to program itself; password-protected access to fields; multiple levels of password protection; write protection.
Applications development. Macro/procedure capability; creation of custom menus; menu system can be embedded in applications; fully programmable procedural language; provides links to DOS and Pascal source library; generates turnkey applications; runtime compiler with full development license available.
Compatibility. Data compatibility: program reads and writes fixed length ASCII and comma-delimited ASCII files.
Delivery. Began in November 1981; 19,000 installed.
Price. 8-bit single-user systems: $750; 8-bit multiuser systems: $995; 16-bit single-user systems: $995; 16-bit multiuser systems: $1,250; XENIX version: $1,250 (available July/August); source code/library: $400.
Runtime licenses. 8-bit single-user system: $200; 8-bit multiuser system: $300; 16-bit single-user system: $200; 16-bit multiuser system: $300. For Canadian licenses, multiply prices by 1.5.
Support. Telephone support; demonstration diskette; end-user training by vendor and distributor; sample applications files; newsletter; bulletin board service. $100 for demo license; similar to full license except that a limit of 25 records per file is imposed; service contracts will be available in the fourth quarter of 1985. Variable fee for updates depending on revisions; last update cost $60 for new commands and complete new manual. Program is not copy-protected; unlimited copies permitted.

This material is taken from Data Decisions Microcomputers, *a three-volume monthly updated reference service that evaluates microcomputer systems, software, and peripheral products. For additional information contact Data Decisions, 20 Brace Road, Cherry Hill, NJ 08034. 609/429-7100*

The benchmark programs provide more of a challenge than the inquiries and reports. The first adds 900 records to an empty database structure from an ASCII text file. The program to do this is essentially that generated by READ to load the database. Only start and stop timing points need to be added.

The results are shown in table 1. The table presents the times observed by *PC Tech Journal* on a PC/AT with a freshly formatted fixed disk. By themselves, these numbers mean very little. When subsequent articles in this DBMS series are published, the benchmark tests will serve as an effective measure of the relative performance of each product. Judging from these times alone, DataFlex's performance seems to be very fast, especially in screen I/O.

The second benchmark is as simple as the first. It creates an index for the database simply by using the REINDEX program. Because the idea of the benchmarks is to time how long applications developed for end users take to perform various tasks, the REINDEX program is run from a DataFlex program. If REINDEX is run as a stand-alone program, it conducts a screen dialogue to select the database, index number, etc.—an end user might not necessarily know about these items. When REINDEX is run from a DataFlex program, these values can be passed as parameters. REINDEX is directed in its command to chain back to a DataFlex application to complete the timing.

The third benchmark is a challenge. The task is to document the codes used in one field of the database and to count their occurrences. The usual method of stepping through a database with DataFlex is to use the REPORT program structure. In this case, a report is generated with QUERY, then modified to add counting logic. This produces an elapsed time of 3:05 on the XT. Skeptical of the efficiency of detailed processing in REPORT, the programmer modifies the code to move the counting logic to a break subtotal, something not provided by the QUERY-generated program. This improves the time to 2:59. Finally, the REPORT structure is discarded in favor of stepping through the file with a loop. The elapsed time drops to 2:35. This program demonstrates many of the routine DataFlex commands, and its essential parts are shown in listing 5.

Once the third benchmark is complete, the fourth is easy because it is quite similar. Its job is to change one field in the database from one value (presumably an error) to another (its correct value). Unlike the third benchmark, which processes the database in index order, the fourth must process it in record sequential order, because an index field is being changed (this can cause records to be skipped otherwise) The challenge is to get DataFlex to process a file without using an indexed order, yet still update the index file. DataFlex has only one mechanism to position within a database: the FIND command. But FIND requires an index to be specified. The solution to this puzzle is a special form of index, the pseudo-index RECNUM. Changing the statement FIND GE author BY INDEX.2 to FIND GE author BY RECNUM accomplishes this task. Another secret is to CLEAR the database buffer prior to the first FIND, thereby forcing the first record to be found as a result.

The fifth benchmark extracts selected records from the database to create an ASCII text file in a particular order. This is a database unload and is the reciprocal function of the first benchmark. What is different is that only selected records are output, and an index order is used. QUERY is ordinarily used to unload a database, but again the benchmark requires no special knowledge of the system, so a program has to be written. QUERY generates the program, but the last field of the database, a 200-character field, is truncated to 80 characters in the QUERY-generated output format.

At times the DataFlex language is verbose and requires several steps to accomplish what other languages would do in one.

To get around this problem, the third benchmark program is merged with the first, then the direction is changed to be an unload. Borrowing a tip from the QUERY-generated program, code is included to "jump in" to the database index at the first selected record and to "jump out" when the selection criteria are no longer met. The result is an accurate, quick extract program.

DataFlex seems to be a product that would meet the needs of most applications developers. Applications programming is the art of bringing available solutions to new problems. A computer language or a DBMS reflects those predefined solutions. To use them effectively, a developer must learn to think like the creators of those solutions and to approach new problems from the point of view of the available solution methods.

The language aspect of DataFlex scores well. One essential meaning of language is that symbols can be placed in syntax structures in such a way as to create new meanings. With the nervous anticipation of unknown problems that the future will bring, applications developers seek out languages with robustness—the ability to express one idea in a many diverse ways.

At times the DataFlex language requires several steps to accomplish what other languages would do in one. However, its abilities to express control, to handle multiple files, each with multiple indexes, and to provide user-directed software actions are strong.

The nature of DataFlex encourages modular application designs. The menu structure lends modules independence from each other, and the data file design forces not just compatibility but single-mindedness in the storage of data. Because opening a file automatically opens all of its indexes, the indexes stay updated along with the file.

There is a cost to obtaining some of these benefits. Binding time must be considered as well as program/data independence. DataFlex applications reference data fields by name, and the compiler resolves each name to a file number, field number, and field type at compilation time. In order to gain the efficiency of runtime speed, a penalty is paid in development labor and discipline. If a database design requires a field to be moved, all programs using that database must be recompiled. Considering the trial-and-error nature of database design, this aspect of DataFlex can be punishing for the developer. Such problems can be anticipated with FILEDEF and record space can be allocated in excess of actual needs. Then, by adding fields only at the bottom of the list, the developer can avoid the need for recompilation or database export/import maintenance. Although some developers may be experts at designing systems, few end users are experts at specifying their requirements.

In general, a relational database system allows for the creation of one database as a subset of another and for the combining of databases with common fields. A relational DBMS also allows a single operation to encompass an entire file, such as changing one

code in a field to another for a selection of records (see benchmark 4). Although it supports related files beautifully, DataFlex is not based on a true relational database model and cannot perform operations on the entire file. Without this feature, DataFlex requires that a program be written to perform changes to or manipulations of the database. The end user of an application does not need this DBMS capability, but the developer must have it if he is to use his time effectively and efficiently.

In addition, many points within the development process demand that the DataFlex programmer have an intimate knowledge of file numbers, field numbers, index numbers, and field offsets within records. When an index is created, for example, it must contain numbers for each item in the REINDEX command. This does not exemplify program/database independence. With DataFlex, the developer must carefully

> *DataFlex's claim is to provide serious applications developers with an effective system for creating applications that, once developed, run on diverse single and multiuser microcomputers.*

consider the implications of adding or deleting database fields, databases themselves, and indexes. Each of these changes may affect programs, other fields within the same database, related databases, and database indexes, because DataFlex relies on an underlying system of number-based relationships. It does not do the bookkeeping required to hold complex relations intact, so the burden of the system's integrity falls onto the developer.

Data Access Corporation makes no claim, however, that DataFlex is a productivity tool. Its claim is to provide serious applications developers with an effective system for creating applications that run on diverse single and multiuser microcomputers. DataFlex is a mature product that reflects that goal.

It is fairly easy to learn, given the AUTODEF and QUERY utilities. The applications developed for this article are reliable and quick. The language is sufficiently sophisticated to deal with complex applications requirements. DataFlex deserves a favorable consideration as a development system.

Chris Christian is a software author and consultant specializing in dBASE applications.

LISTING 1: AUTHOR.SRC

```
                    AUTODEF GENERATED PROGRAM FOR AUTHOR DATABASE

/MYSCREEN
|------------------------------------------------------------------------------|
|                      EDITORIAL INVENTORY -- Author File                      |
|------------------------------------------------------------------------------|
|                                                                              |
|    Author: __________________ ____________ Work Phone: (___)___-____         |
|            ----Last name----  -First name- Home Phone: (___)___-____         |
|                                                                              |
|    Social Security Number ___-__-___                                         |
|------------------------------------------------------------------------------|
|   Address: ____________________                                              |
|      City: ________________        State: __  Zip: _____                      |
|------------------------------------------------------------------------------|
| Biography: __________________________________________________                |
|            __________________________________________________                |
|            __________________________________________________                |
|            __________________________________________________                |
|                                                                              |
|------------------------------------------------------------------------------|

/*
PAGE MYSCREEN
OPEN AUTHOR
ENTER AUTHOR
AUTOPAGE MYSCREEN
  ENTRY AUTHOR.LAST_NAME
  ENTRY AUTHOR.FIRST_NAME
  ENTRY AUTHOR.WORK_PHONE
  ENTRY AUTHOR.W_PHONE2
  ENTRY AUTHOR.W_PHONE3
  ENTRY AUTHOR.HOME_PHONE
  ENTRY AUTHOR.H_PHONE2
  ENTRY AUTHOR.H_PHONE3
  ENTRY AUTHOR.SOC_SEC_NUMBER
  ENTRY AUTHOR.SS_NUM2
  ENTRY AUTHOR.SS_NUM3
  ENTRY AUTHOR.ADDRESS
  ENTRY AUTHOR.CITY
  ENTRY AUTHOR.STATE
  ENTRY AUTHOR.ZIP
  ENTRY AUTHOR.BIO1
  ENTRY AUTHOR.BIO2
  ENTRY AUTHOR.BIO3
  ENTRY AUTHOR.BIO4
  RETURN
ENTEREND
ABORT
```

LISTING 2: AUTHOR2.SRC

```
                ENHANCED PROGRAM TO MAINTAIN THE AUTHOR DATABASE

/HELP

        Press F9 to clear the fields before locating a new record.

    Fill in (or start) the author's name, then press F3 to find a record.

        After a record has been located,
              press PgUp or PgDn to skip forward or backward records.

                 <Press RETURN to resume maintenance>
/*
STRING valid_states 150
MOVE "AK|AL|AR|AZ|CA|CO|CT|DE|FL|GA|HI|IA|ID|IL|IN|KS|KY|LA" TO valid_states
APPEND valid_states "|MA|MD|ME|MI|MN|MO|MS|MT"
APPEND valid_states "|NC|ND|NE|NH|NJ|NM|NV|NY|OH|OK|OR|PA|RI"
APPEND valid_states "|SC|SD|TN|TX|UT|VA|VT|WA|WI|WV|WY"
PAGE myscreen
OPEN author
ENTER author
AUTOPAGE myscreen
  ENTRY author.last_name {autofind,skipfound}
  ENTRY author.first_name {autofind,skipfound}
  ENTRY author.w_phone1 {points=0}
  ENTRY author.w_phone2 {points=0}
  ENTRY author.w_phone3 {points=0}
  ENTRY author.h_phone1 {points=0}
  ENTRY author.h_phone2 {points=0}
  ENTRY author.h_phone3 {points=0}
  ENTRY author.ss_num1 {points=0}
  ENTRY author.ss_num2 {points=0}
  ENTRY author.ss_num3 {points=0}
  ENTRY author.address
  ENTRY author.city
  ENTRY author.state {capslock,check=valid_states}
  ENTRY author.zip {points=0}
  ENTRY author.bio1
  ENTRY author.bio2
  ENTRY author.bio3
  ENTRY author.bio4
```

```
   RETURN

ENTER.SAVE:
// any special pre-save editing can be placed here
RETURN

ENTEREND
ABORT
```

LISTING 3: AUTHOR.FD

```
                    AUTHOR FILE DEFINITION

        FILE DEFINITION LISTING FOR FILE #1
*****************************************************
        FILE ROOT NAME      = Author
        USER DISPLAY NAME   = Author
        DATAFLEX FILE NAME  = AUTHOR
*****************************************************
        RECORD LENGTH = 384   (USED = 302)
        MAX NUMBER OF RECORDS = 900   (USED = 903)
        DELETED SPACE IS REUSED
        MULTI-USER RE-READ ACTIVE
*****************************************************
FIELD  FIELD   FIELD   FIELD   DEC  MAIN   RELATES--TO
NMBR   OFFSET  LEN     TYPE    PTS  INDEX  FILE  FIELD
-----  ------  -----   -----   ---  -----  ----  -----
  1       1     18     ASCII         1      0     0     LAST_NAME
  2      19     12     ASCII         1      0     0     FIRST_NAME
  3      31     20     ASCII         0      0     0     ADDRESS
  4      51     16     ASCII         0      0     0     CITY
  5      67      2     ASCII         2      0     0     STATE
  6      69      5     ASCII         2      0     0     ZIP
  7      74     10     ASCII         0      0     0     WORK_PHONE
  8      84     10     ASCII         0      0     0     HOME_PHONE
  9      94      9     ASCII         0      0     0     SOC_SEC_NUMBER
 10     103    200     ASCII         0      0     0     BIOGRAPHY
 11       1     30     OVERLAP       1      0     0     NAME_KEY
 12      74      3     OVERLAP       0      0     0     W_PHONE1
 13      77      3     OVERLAP       0      0     0     W_PHONE2
 14      80      4     OVERLAP       0      0     0     W_PHONE3
 15      84      3     OVERLAP       0      0     0     H_PHONE1
 16      87      3     OVERLAP       0      0     0     H_PHONE2
 17      90      4     OVERLAP       0      0     0     H_PHONE3
 18      94      3     OVERLAP       0      0     0     SS_NUM1
 19      97      2     OVERLAP       0      0     0     SS_NUM2
 20      99      4     OVERLAP       0      0     0     SS_NUM3
 21     103     50     OVERLAP       0      0     0     BIO1
 22     153     50     OVERLAP       0      0     0     BIO2
 23     203     50     OVERLAP       0      0     0     BIO3
 24     253     50     OVERLAP       0      0     0     BIO4

INDEX 1:  FIELD SEGMENTS:  <11>
INDEX 2:  FIELD SEGMENTS:  <5>   <6>   <0>
INDEX 10:  FIELD SEGMENTS:  <5>    BATCH INDEX
```

LISTING 4: REPORT1.RPT

```
                 QUERY GENERATED REPORT PROGRAM

/HEADER
            PC Tech Journal Editorial Listings -- Report #1

TITLE                                                EDITORIAL LISTING
  AUTHOR-------------------  CO-AUTHOR-----------------  PAGES   PAGES

/BODY
______________________________________________________________________
__________  __________  ____________  __________   __.   __.
/TOTAL
EDITORIAL PAGES    __________.__
LISTING PAGES      __________.__
RECORDS PRINTED =  ______.
/SELECTION
```

```
            PC Tech Journal Editorial Listings -- Report #1

Select records by VOLUME equal to __.
Select records by NUMBER equal to __.
/*
OPEN ARTICLE INDEX.1
OUTFILE "CON:"
  PAGE SELECTION
  ACCEPT SELECTION.1
  ACCEPT SELECTION.2
REPORT ARTICLE BY INDEX.1
              INDICATE SELECT AS ARTICLE.VOLUME EQ SELECTION.1
[    SELECT] INDICATE SELECT AS ARTICLE.NUMBER EQ SELECTION.2
SECTION HEADER
        OUTPUT HEADER
SECTION BODY
        PRINT ARTICLE.TITLE
        PRINT ARTICLE.AUTHOR_L_NAME
        PRINT ARTICLE.AUTHOR_F_NAME
        PRINT ARTICLE.CO_AUTH_L_NAME
        PRINT ARTICLE.CO_AUTH_F_NAME
        PRINT ARTICLE.EDITORIAL_PAGES
        PRINT ARTICLE.LISTING_PAGES
        OUTPUT BODY
SECTION TOTAL
        SUBTOTAL BODY.6 TO TOTAL.1
        SUBTOTAL BODY.7 TO TOTAL.2
        PRINT RECCOUNT TO TOTAL.3
        OUTPUT TOTAL
REPORTEND
FORMFEED
ABORT
```

LISTING 5: BENCH3.SRC

```
                     BENCHMARK 3 PARTIAL LISTING

/* Bench3B.Src
STRING last_state 2                      // Our state teltale
MOVE 'xx' TO last_state                  // Its initial value
INDICATOR same_state                     // TRUE if state is same as last
INDICATOR next_one                       // FALSE initially, then true
INDICATE next_one FALSE

NUMBER count                             // Tally of occurrences, this state
MOVE 0 TO count                          // Initial value

NUMBER n_states                          // Number of states
NUMBER total_count                       // Record count

OPEN AUTHOR INDEX.2                      // Open the author file by state+zip
CLEAR AUTHOR                             // Clear the record buffer

CLEARSCREEN
FIND GE author BY INDEX.2                // Find the first record
[FOUND] REPEAT                           // If any records, begin loop
   INDICATE same_state AS author.state MATCH last_state
   [same_state] CALC (count+1) TO count
   [NOT same_state] BEGIN
            [next_one] BEGIN
                   SHOWLN last_state " = "count
                   END
            INDICATE next_one TRUE
            CALC (n_states+1) TO n_states
            MOVE author.state TO last_state
            MOVE 1 TO count
            END
   CALC (total_count+1) TO total_count
   FIND GT author BY INDEX.2             // Find the next record
[FOUND] LOOP                             // If more, loop back

SHOWLN last_state " = " count
SHOWLN
SHOWLN "Number of states = " n_states " Number of records = " total_count

ABORT
// End of program.
```

IF YOU PROGRAM IN 'C', PC BRA

Our Craftsman™ Line Has New Products,

C-SPRITE

Lattice's Own Symbolic Debugger for Lattice® C

This versatile companion to your compiler gives you the best of both worlds for an out of this world price. Hand it a COM or EXE file produced by the Lattice compiler (using the -d option) and C-Sprite™ will speak your language: your function names, your variable names, your data types, and the line numbers from your source code. At the same time you can get a close-up view of machine addresses and machine-coded instructions, if you want to scrutinize just what machinations the compiler (or an assembler) contrived.

You already know how to converse with C-Sprite if you are familiar with Microsoft's Debug. Lattice began with that well-known command language, and added to it considerably: You can work with data in hex, as you might expect, but you can also differentiate between C's data types to cause the debugger to treat addresses as strings, long integers, etc., even pointers, both in display and entry.

C-Sprite can set breakpoints using symbols or addresses. You can submit clusters of commands to be executed at the breakpoints, or set commands that execute until a condition is met.

C-Sprite even has macros—use your source code variable names in a macro to dump the contents of entire C structures, for example. And you can debug through one of the COM ports with a second terminal so as not to disturb your program's display screen. What's more, if you link with Plink86™, C-Sprite can even tackle overlays.

Program doctors will find plenty of implements to rummage through in this kitbag.

Product Code:	List Price:	Our Price:
L2300	$175	$159

EXPORT EXPERTISE

PC BRAND ships anywhere. We'll prepare the export documents and ship to you or your agent by air parcel post, air freight or courier. Phone or Telex your order. Pay by credit card or wire funds to PC BRAND, % Chemical Bank, 126 East 86th St., New York, N.Y. 10028, Account No: 034-016058. We will ship immediately.

TEXT TOOLBOX™ #1

These Utilities Work Wonders of Organization

Welcome to "grep", "wc", "ed" and "diff", tools you will reach for as routinely as "copy" once you come to know them. Unix™ boasts a number of muscular utilities that are migrating to the PC world. Lattice has assembled a cluster of the most useful text management tools into a single package.

"Grep" looks for text patterns in any number of files. Want all occurrences of a global variable throughout a program system? Want to search all programs in a directory, down paths to other directories, or all files on a disk? Need to find all the function calls in an entire program system? Grep can do it with a powerful expression syntax that goes far beyond your text editor's search command, because you can tell it to search all "*.c" files for all lines with "(" and ")", no matter how many characters lie between the parentheses. In text searches it will match any character in whatever character range you cite, in a single character position or anywhere in a string, as you specify. Or "Grep" will match patterns only at the beginnings or endings of lines, and can differentiate between embedded and isolated strings. "Grep" is a real grabber.

"WC" counts lines, words, and characters in a file and has a checksum independent of machine character sets so you can test whether a file has successfully been transferred between computers.

"Ed" is similar to the well-known Unix editor. It offers search and replace with "Grep's" syntax, block move, read and write, optional line numbering, append, insert, delete, and this unusual facility: you can instruct "Ed" to apply a file of commands to any number of target files, even complicated changes and text additions, such as those created by "Diff".

"Diff"? You've probably tried to write one (and then discovered how tangled the logic gets). "Diff" compares text files line for line and reports differences. It's much more sophisticated than "filecom", if your MS™-DOS has that. It can optionally ignore "white space" differences (blanks, tabs). It uses complex algorithms to re-synchronize between files after disparities of any number of lines are found. And its output is a precise list of instructions telling what to do to make two files the same, a list which you can hand to "Ed" to do the job!

You'll ultimately find such assistance indispensable. Like having a librarian to sort out the confusion every day and keep your work tidy.

Product Code:	List Price:	Our Price:
L2200	$120	$105

CURSES

A Screen Management Interface to Swear By

Curses is a Lattice creation which manages the screen of the IBM PC in the same fashion as the curses utility of Unix and similar operating systems. Use it to adapt programs which call Unix's curses functions for screen management, and need the equivalent library when moved to the PC for re-compilation. Or use it when creating software on the PC to assure that it is Unix compatible.

Curses is a library of eighty-four functions and macros which can keep any number of screen images in memory. A screen may be full or partial size, and any can be summoned to the physical screen at your programs's command.

Within a screen, Curses employs a vast function set to get characters, wrap lines, scroll, blank lines, highlight—virtually any tool needed to update the screen. The product supports color, and all four memory models. Its input functions give you control over whether to echo each character to a memory screen. In keeping with the terminal orientation of Unix curses, the physical screen is repainted (at high speed) only when your program calls a refresh function.

Writing screen management code leads to unspeakable snarls and expressions. Swear off! Let Curses clean up your language.

Product Code:	List Price:	Our Price:
L0850	$125	$110

C CROSS COMPILERS

Portability to 16-Bit With Cross Compilers From Lattice

PC BRAND now carries an assortment of cross compilers to move products from larger host machines to the PC-DOS or MS-DOS environment.

Cross compilers are now available for these hosts: VAX/VMS™, VAX/UNIX, MC68000/UNIX, Altos 586, Hewlett Packard-UX.

Cash in on products already developed on your bigger machines by rapid transfer to the burgeoning world of PC owners. Or take advantage of big machine services and utilities for your development work for the PC market, and only then download the results. A quick mental calculation will convince you that productivity gains will quickly trade off costs.

Expand Your Sales to the CP/M World

They have vanished from the headlines, but there are over a million CP/M™ machines still humming across the land. Yet *Infoworld* reports that "there's no new CP/M software". If you are part of the problem, here's a money-making way to become part of the solution. Convert your MS-DOS/PC-DOS products with our CP/M-Z80 targeted cross compiler. For a few dollars, you'll double your market in a hurry.

CODE SIFTER

Find the Fast Track for Your Program

You know it runs, but how fast? If it can't win, better not enter it in today's horse race. Don't let it out of the stable until you've put it through its paces with this perfectly priced stopwatch.

Code Sifter finds the sluggish spots in your program. On its own, it will divide a COM or EXE file of any size into thirty-two equal partitions. Alternatively, you can specify the partition boundaries with addresses, or with symbols if your linker has produced a symbol map. Then tell Sifter to run your program. It samples your object file at precisely timed intervals and counts how many times it finds the instruction pointer in each partition. Job done, it reports the number and percentage of hits in each partition.

You are in for some surprises when you discover just how unbalanced the activity is likely to be, and that's why Code Sifter is so valuable. It profiles just where you can best spend time optimizing your code, or even converting to assembler subroutines.

Code Sifter has a number of monitoring options. You can tell it to include any combination of your program, DOS, and BIOS in its analysis. You can specify the sampling rate. You can tell Code Sifter the number of times to run a program, and between each run discard the less active ranges, and re-partition the hot spots, so that you zoom in ever tighter on small areas of code. Right down to the last byte! Try it on the sample program that comes with your disk.

Code Sifter. It will give you the racer's edge.

Product Code:	List Price:	Our Price:
N3100	$119	$99

CVUE

A Text Editor to Make Your Own

CVUE is a neat screen oriented text editor which does most of the things that a good editor should do, such as automatic scrolling vertically and horizontally, insertion and overtype entry modes, block delete, undelete and move, and full DOS 2.0 directory path name support in reading and writing files.

It is easy to learn with a comprehensive command menu screen which makes the documentation an ornament. It was written by the Lattice programmers who felt forgotten by the folks who write WP software. They needed easy entry of non-display characters such as control codes and escape sequences, not footnotes. Indenting and undenting of block structures loomed larger than italic printing for them. Pattern searching won out over spelling checking. So CVUE was born.

CVUE has its limitation. It only supports in-memory text files, but with memory at today's prices, creating and maintaining files 500 KBytes long is practical. Anyway, modular source code of structured programs never gets nearly that big. As compensation, CVUE is very compact and fast. Even in 64K computers it has no need for tediously slow overlays to perform its full function repertoire.

The power of CVUE is its ease of customization. Even the binary version offers full customization of the keyboard editing commands. And when you take advantage of the Source Code option on the next page, the resultant editor can be made truly your own.

Product Code:	List Price:	Our Price:
L2240	$100	$90

dBC

Switch from dBASE's Language to C for Power, Speed

There are a lot of dBASE™ file users out there. Most of them just keep data bases and use dBASE's limited reporting facilities. They're not programmers, so they don't use the dBASE programming language. But they'd like more for their efforts, and that's a business opportunity.

dBC™ links C to dBASE. It is a function library that creates files which exactly replicate dBASE file design. So dBASE can read and update them. And the reverse. dBC can use any files created by dBASE. Now C and dBASE can operate on the same data bases interchangeably.

That opens up the widespread culture of dBASE installations to exploitation by C programmers. Now you can replace the resident dBASE language with the speed of C. And you no longer have to write every line of code, because moving to C unlocks C's vast storehouse of off-the-shelf libraries and utilities.

Use dBC for custom work for customers, or design generalized programs for manipulation and reporting of dBASE data bases. Or use it on its own. It's a complete ISAM file manager for use with the Lattice C compiler whether or not dBASE will ever be used in tandem, has versions for all four memory models, and can have sixteen index and data files open at once. dBC is a complete set of ISAM routines (that parallel dBASE commands) which the manual and demonstration source files on the disk explain in detail.

Two Versions:	Product Code:	List Price:	Our Price:
Dbase II Compatible:	L00II	$250	$225
Dbase III Compatible:	L0III	250	225
Both:		500	375

ND™ SPEAKS YOUR LANGUAGE

Reduced Prices & Plenty of Source Code

C-TREE

B-Tree File Manager, Source Code, No Royalties! NEW!

A b-tree can be infested with bugs, so before buying one, ask its age. In a stand of saplings, this one is a real cequoia. C-tree™ has been around since 1979. That means seasoned, sturdy code which hasn't cracked under the weight of prolonged and widespread use.

Its developer has added two very unusual features. C-tree now comes in C source code, revealing all you've ever wanted to know about how b-trees are written. Second, provided you bind it into your binary application, you can re-distribute C-tree without royalties.

Thanks to source code which does not deviate from the K&R standard, C-tree can travel. Binary has always meant finding a substitute file manager when you port your work to another compiler, operating system, or computer; then changing all the function calls and testing anew. That's over. Tests in many environments prove that C-tree gives your application a ticket to anywhere.

C-tree permits any number of keys for a data file, supports duplicate keys, alphanumeric or numeric, etc., etc.: it's a big product with everything you'd expect. Beyond that it is intelligently designed as both a high level set of ISAM routines to minimize your coding by handling all details of adding a record on its own, for example; and as low level operations which you can access directly. Either way C-tree maintains optimal index structures which will find a record amongst a million ten byte keys in no more than five disk seeks.

And if all this is disappointing, now the good part. C-tree's design splits nodes to allow any number of users to access an index file simultaneously even when updates are in progress so that multi-user configurations and adaptation to networks are possible. You must write the record-locking routines, as they are compiler and operating system dependent, but the documentation shows how by example.

Product Code:	List Price:	Our Price:
F0660	$395	$345

LMK

A Unix-like "Make" Makes Light of System Building in Any Language

If you have ever built a complex system, you know the time loss and tedium of recompiling, rebuilding libraries and relinking modules because a snippet or two of code has changed. Batch files are no answer. You need batches of them to avoid redoing everything indiscriminately.

Instead, imagine making a change deep in a system, and simply telling Lattice's LMK™ to take over. No further thinking or keystrokes. LMK will rebuild your final product, however involved and complex, by doing just what is needed and no more.

How? You write a command file which expresses, bottom to top, all the elements comprising your system and all its dependencies: what gets compiled to make what object file using what options; what is built into libraries; what is to be linked into the final EXE file. Through the life of your system LMK keeps track of the last time every action was performed. Run LMK and, tracking each branch, it looks only for elements which changed later than a dependent element further along the branch, using date and time information in the file directory. Any source file newer than its object file, for example. Only those elements and their dependents are re-made. All other instructions are bypassed.

The command file uses a simple, readable syntax—"prog.obj: prog.c $(HDRVS)", for example, says what source file this object file depends on, and says to fill your previously-defined macro HDRVS into the expression, which in this case might be a list of files with hardware drivers, or in another case your preferred string of compiler options.

LMK does not care what programming language you use; it's not just for C. For that matter, LMK can apply to more than programming. It can be used for *any* set of tasks which can be accomplished through commands issued to the operating system. Try it for repeated reassembly of lengthy documentation, or for *selective* reconsolidation of spreadsheets so that only the dependents of altered supporting schedules get re-calculated.

Wherever your imagination leads you, LMK will find the shortest path to get the job done. Minimum time, minimum effort software.

Product Code:	List Price:	Our Price:
L2100	$195	$175

USED COMPILERS WELCOME

Trade In for the Latest Model Lattice C

Has your compiler run out of gas? Has your model been discontinued? Is it falling behind for lack of new parts?

Even if yours is in good shape, you have surely noticed there are more options and accessories produced to run with the Lattice C compiler than any other. Don't do without these additives any longer. It's time for new license plates. Trade in your original disks and manual of any of the compilers below and we'll send you Lattice's most up-to-date model, the full bore 2.15 (at press time).

From then on you will be adopted by Lattice for full, direct support by their technical specialists.

	Price:
Microsoft MS-DOS/PC-DOS C	$150
Computer Innovations C86, Wizard C, Manx Aztec C86, Mark Williams C, Digital Research C, Whitesmith's C	$200

PRE-C

Thorough "Lint"-like Analysis Now on the PC NEW!

Unix users have long asked if we had a "lint" to give programs a thorough cleaning before they disappear into a compiler. The answer now is "yes".

Pre-C™ finds problems your compiler won't. Problems that a debugger will have trouble figuring out. Even problems which will cause trouble with other compilers.

Pre-C finds all the syntactical tripwires that will blow out a compile, sure, but it goes after subtler problems: code which will never be accessed, casts with suspect conversions, variables declared as external but never used, functions never called, obsolete usage (even the C language has changed), machine-dependent expressions which will inhibit portability.

Compilers work with one module at a time. They know nothing of other modules which only meet up at link time. Pre-C can look at *all segments of your program at once* and report to you any inconsistencies of inter-module references: conflicting data type declarations, parameter lists in function calls which disagree with the functions themselves in number or data type, declarations of external functions which differ from their definition.

Pre-C uses the Unix System III compiler standard to safeguard maximum portability anywhere in the C world. There are then plentiful command line options to advise Pre-C what to flag and what to forgive, useful during early coding when some functions are empty or incomplete. The resulting analysis can be filed for use with subsequent Pre-C runs, so work is not performed redundantly.

Pre-C lets you develop standing profiles of binary libraries you have purchased. In any C program you subsequently write, Pre-C can use these profiles to make sure your calls to those libraries' functions are perfect.

This is a big product which will work miracles in speeding large system development.

Product Code:	List Price:	Our Price:
P0590	$395	$329

Normally fits in 128K but requires 192K to analyze large programs. Requires disk space about 20% the size of all files to be analyzed.

SUPPORT DIRECT FROM LATTICE

Products with codes beginning "L" are all fresh from Lattice Inc., the premier software developer serving the C professional marketplace. PC BRAND has made special arrangements for Lattice to provide telephone support for these products directly.

Take advantage of this unusual opportunity. Buy from PC BRAND, and you get to talk to the experts themselves!

30-DAY MONEY-BACK GUARANTEE[a]

It gets better. We will return the purchase price[b] of any "L" or "N" coded product offered in this space if you decide it fails to meet your needs. We want you to be completely satisfied with your purchase. Go through the manuals, *try out the products themselves*, and make a thoroughly informed decision whether they are right for you.

There are rules: We can make this offer in the U.S. and Canada only. To be eligible for refund or credit, we must *receive* your returned product within 30 days of its shipment, and it must in our judgment be in 100% resaleable condition.

(a) Subject to terms and conditions of sale.
(b) Not shipping and handling.

SOURCE CODE FOR SALE

Designer Originals For Your Special Needs

"One size fits all" turns away disappointed customers. When you cannot shoehorn your application into out-of-the-box software, we have the solution. For each of these products only, PC Brand now licenses source code, provided you buy (or have already bought) from us the object code counterpart. Take it in, let it out, and make it fit your needs snugly. And, if you are new to C, you will learn a lot apprenticing this fully documented code from top designers.

	Product Code:	List Price:	Our Price:
Lattice C Compiler Library			
for MS-DOS/PC-DOS	L9100	$500	$450
for Z80	L910Z	500	450
C-Food Smorgasbord Library	L9200	500	450
Curses Library	L9850	125	110
dBC Library for dBase II	L90II	250	225
dBC Library for dBase III	L9III	250	225
CVUE Screen Editor	L9240	350	315
Text Toolbox #1	L9220	120	105

TERMS AND CONDITIONS OF SALE

Licenses: Each price is for a license to use a product on a single computer and does not constitute its ownership. Ask about availability of other terms for multiple machine use at a single site. Products may be used to create programs for distribution without royalty payments or additional licenses, provided said programs do not substantially replicate the products themselves.

Compatibility: PC BRAND's standard products are designed to operate with the IBM PC, PC/XT or AT under PC-DOS and require no more than 128K of RAM unless indicated. For non-IBM machines using MS-DOS, determine compatibility from manufacturer.

Returns: Defective parts will be replaced. For products not coded "N" or "L": Acceptance for refund or credit only by prior authorization subject to re-stocking charge. Diskettes in sealed envelopes unreturnable once seal is broken.

Payment: We honor MasterCard & Visa (no surcharge), wired funds, checks in advance, and COD for cash, money order, or certified check (no COD charge). Purchase orders accepted from larger corporations and institutions at our discretion; terms net 30, 2% a month late penalty thereafter.

Shipping & Handling: U.S.: UPS Surface: 1st product $6, each add'l $3. UPS 2nd Day Air: 1st product $10, each add'l $4.50. UPS Next Day Air or Federal Express: 1st product $18, each add'l $6. International: Charges depend on destination and shipping method.

The Programmers Choice.

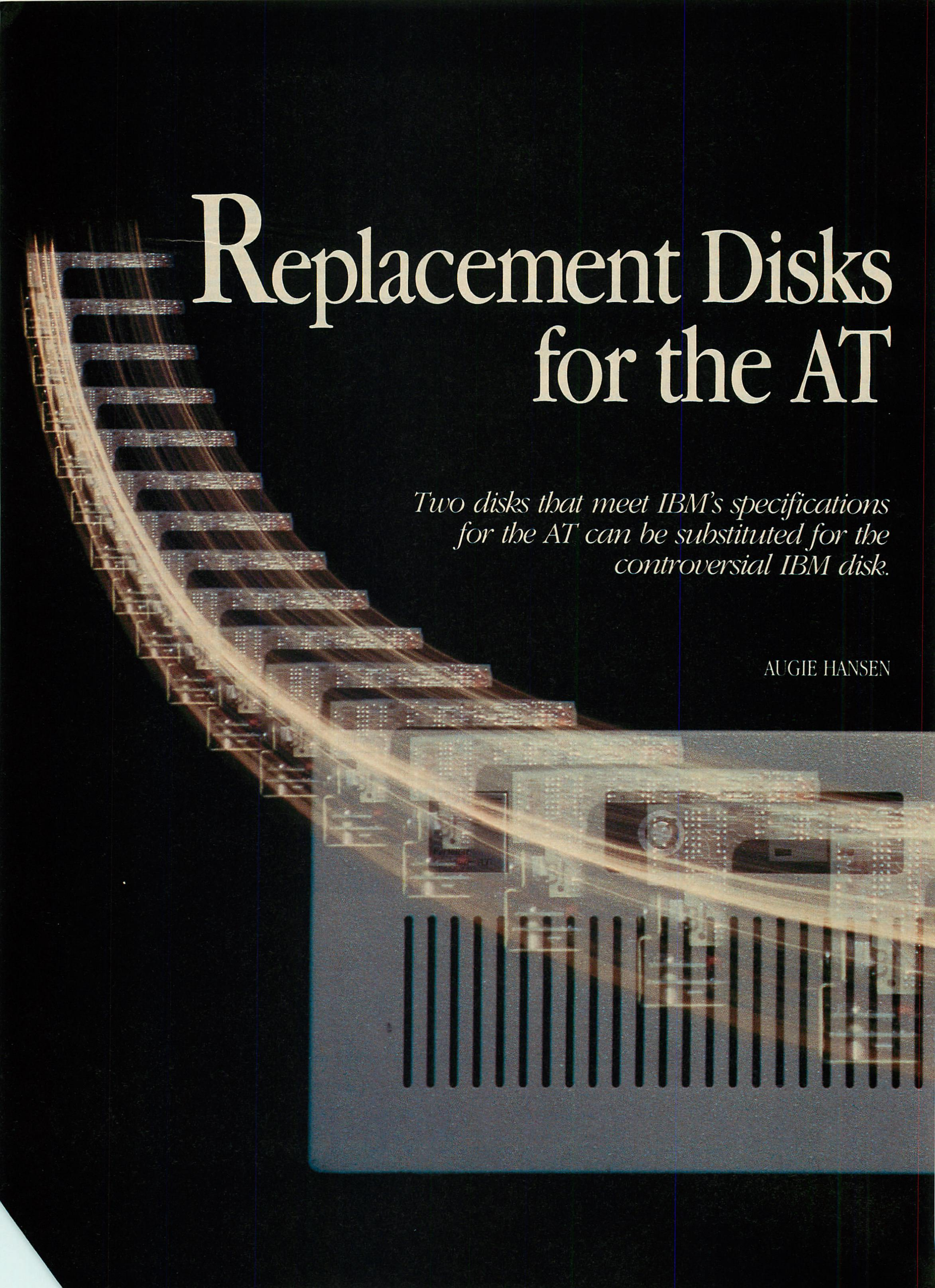
800-PC BRAND

That's (800) 722-7263. In NY State call (212) 410-4000.
PC Brand, P.O. Box 474, New York, N.Y. 10028
Telex: 667962 (SOFT COMM NYK)

Prices, terms, and specifications subject to change without notice.

SAVE $5.00. We'll take $5 off your or[der] if you tell us thi[s]
J8

CIRCLE NO. 171 ON READER SERVICE CARD



Replacement Disks for the AT

Two disks that meet IBM's specifications for the AT can be substituted for the controversial IBM disk.

AUGIE HANSEN

Pioneering PC/AT users like to relate horror stories about fixed-disk crashes, resulting in the loss of many megabytes of precious data. Whether it's through forgetfulness, the thrill of taking risks, or just plain laziness, few AT users back up their fixed-disk files as often as they should. So when that fixed disk begins to shriek its electronic swan song, not much can be done to save the day.

Despite stories to the contrary, the IBM drive, which is produced by Computer Memories, Inc., performs well, if used correctly. CMI has been the victim of some circumstances beyond its control, in addition to experiencing the typical problems associated with carrying out a complex production task at an exponential rate of growth.

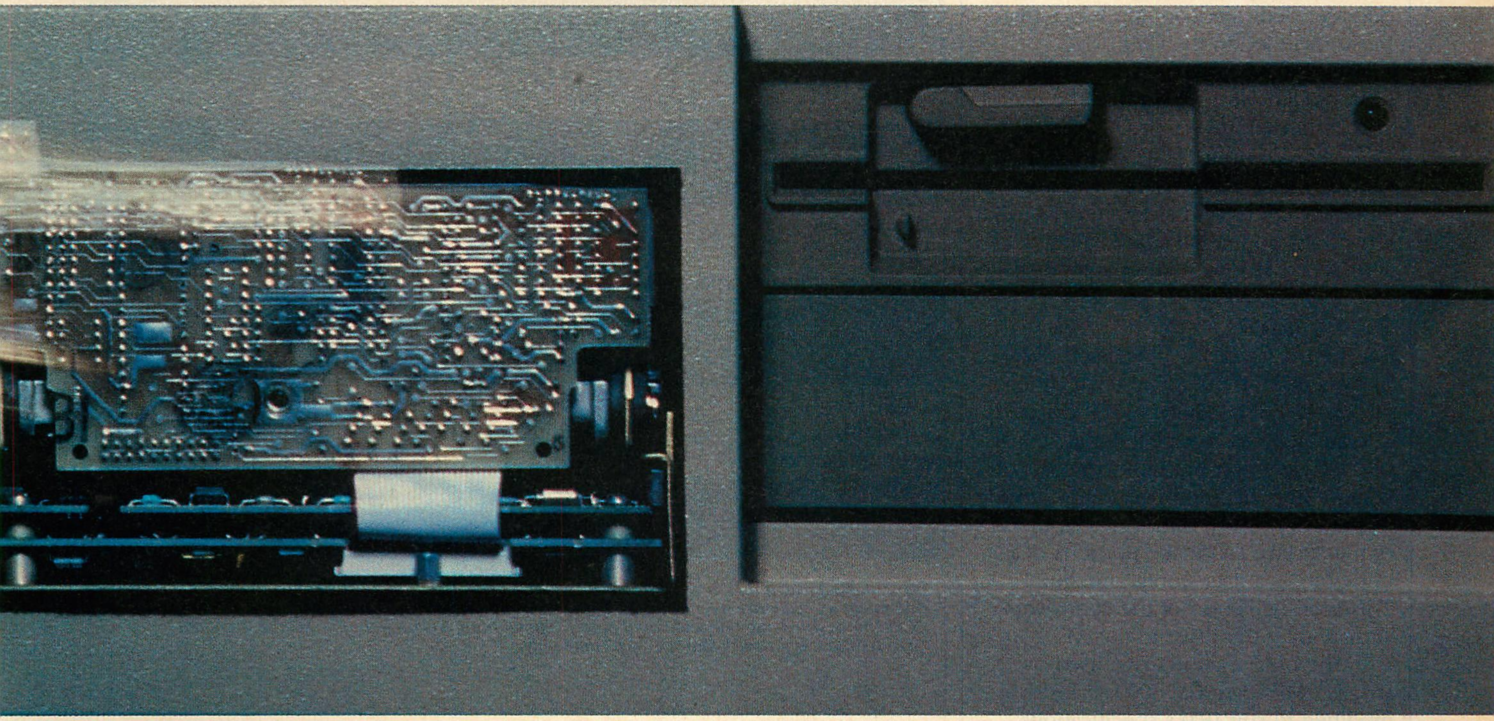
Any AT owner who wants to enhance a base-model machine should be skeptical of many of the fixed-disk drives being offered as AT-compatible—they are simply not up to the task. The major shortcoming of third-party disks is the average access time—that is, the amount of time, averaged over many random and sequential seeks, that is required to find a specified track and sector and to get ready to read or write data. For acceptable AT performance, a disk must have an average access time of 40 milliseconds or less.

At the time that this review is being written, only two products meet the IBM specifications for AT fixed-disk drives: CORE International's ATplus; and Express Systems' 20MB drive. Both CORE and Express Systems use a Control Data Corporation fixed disk that is modified by CDC to meet the AT drive specifications. The drives are altered versions of disks from the CDC WREN series, which are available in 21MB to 86MB unformatted capacities. CDC had to modify the drive in order to bring up the formatted capacity from about 18MB to a little more than 20MB. The CDC drives offered by CORE and Express Systems meet or exceed all of the IBM specifications (see table 1), use the existing Western Digital disk controller, and are very solid mechanically.

The article "Fixed-disk Benchmarks" (William Hunt, *PC Tech Journal*, November 1984, p. 64) provides a good overview of fixed-disk terms, operation, and performance evaluation techniques. The **autotest** program described in that article was used to evaluate the AT disk drives, along with a program called **diskp**, offered by CORE International for $20. Written by CORE to evaluate disks for the company's VAD (value-added dealer) program, it measures access times averaged over 1,000 seek operations, and its results correlate well with observed disk performance and with vendor specifications.

Disk benchmark programs were run on an AT with PC-DOS 3.0 and 640KB of RAM. Tests were run under identical conditions on the CMI and CDC drives, each being formatted for exclusive use by PC-DOS 3.0 in a single 20MB partition. An interleave factor of three was used for all drives.

PHOTOGRAPH • THE STROBE STUDIO



The **autotest** program runs a set of sequential and random I/O operations and reports the results in seconds per read (table 2). The sequential reads are done in 1- , 8- , 16- , and 24-sector blocks. *Widths of seeks* refer to the amount of head movement radially over the working data surface expressed as a fraction of the full width. Thus, a 0.1 width seek spans 61 tracks. At 750 tracks per inch, that's less than one-tenth of an inch of movement.

The **diskp** program reports results in milliseconds for a series of sequential, mixed sequential and random, and entirely random accesses (table 3). The **autotest** program result that most closely approximates the average access time produced by **diskp** is the random single sector read that uses a seek width of 10 percent of the disk's data surface. As can be seen from the tables, all drives performed similarly.

If speed were the only consideration, many drives would have been covered in this article. Speed alone, however, does not guarantee that a disk drive will be a satisfactory substitute for the CMI drive. The CMI drive is identified as type 2 in the AT set-up procedure, meaning that it has 615 cylinders, four heads, and a 21.4MB formatted capacity. Therefore, drives that are not type 2 were excluded from consideration in this review. Drives with capacities other than 20MB have different numbers of disk platters and heads, different precompensation settings, and different dedicated landing zones than those included in the CMI drive.

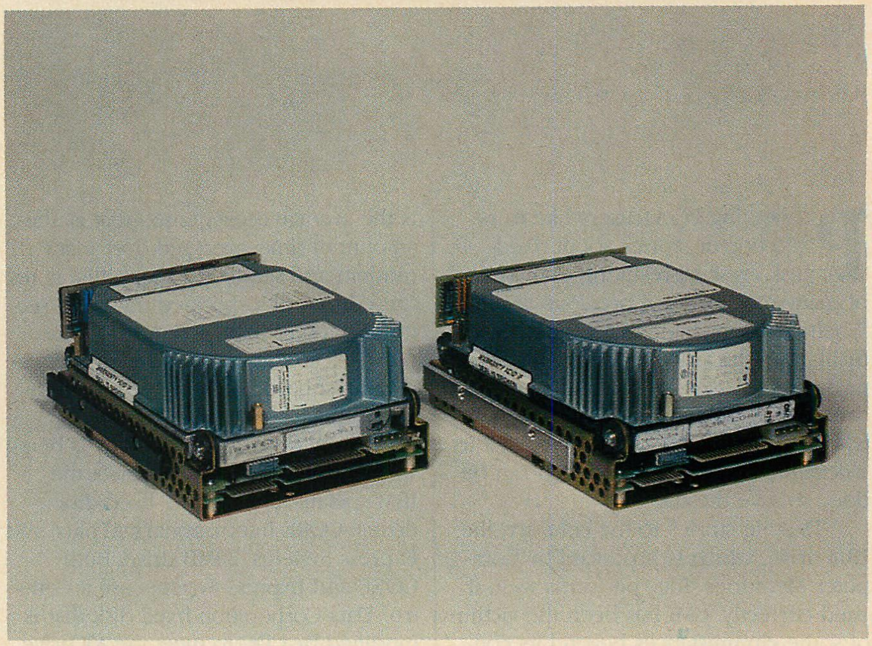
Each of the selected drives is able to boot DOS directly without the use of special device drivers or DOS patches. DOS can use up to 32MB in a single disk partition. Larger capacity drives generally come with a device driver that allows any additional space to be accessed as if it were a separate disk.

DISK BASICS

The CMI drive has oxide-coated disk platters using the Winchester technology of the IBM 3350-type disk drives. The surfaces are not plated; they consist of effectively thin but uniform coats of material in which information is stored magnetically. Even minor deterioration of the surface can make portions of it unusable as a data storage medium. This is at the center of the controversy surrounding the drive in the AT.

The basis for the drive used by IBM is CMI's own CM-6000-series drive, modified to suit IBM's specifications. The CMI drive lacks an automatic mechanism to move the heads off the normal

PHOTO 1: *20MB Add-in Drives*



Both CORE International and Express Systems offer a modified version of Control Data Corporation's WREN disk, which is very solid mechanically.

TABLE 1: *20MB Fixed-Disk Specifications*

	IBM	CORE	EXPRESS SYSTEMS
FUNCTIONAL			
Rotational speed (rpm)	3,573	3,600	3,600
Cylinders (+ landing zone)	615	—	614
R/W heads[a]	6	4	4
Tracks	—	2,460	2,456
Track density (tracks/inch)	750	800	800
PERFORMANCE			
Capacity (MB)			
Formatted	20	21.4	21.79
Bytes/sector	512	512	512
Sectors/track	17	17	17
Transfer rate (megabits/sec)	5	5	5
Access time (ms)			
Track-to-track	2[b]	—	5
Average	40	39	30
Maximum	85	—	70
Settling	12	—	—
Average latency (ms)	8.4	8.33	8.33
MTBF (powered hours)	—	20,000	15,000

Items marked with a dash are not specified by the disk supplier for the disk being evaluated.
[a] Although six heads are specified by IBM, only four are used for reading and writing user data. The CDC-made drives use one additional head for positioning feedback in a dedicated servo loop.
[b] The IBM spec is for track-to-track "step" time only.

Each of the acceptable substitutes for the IBM AT 20MB drive qualify as a type 2 disk in the BIOS disk type table and have an average access time of 40 ms or less.

TABLE 2: *Autotest Results*

	IBM	CORE	EXPRESS SYSTEMS
TOTAL NUMBER OF SECTORS	41,636	41,636	41,620[a]
TIMINGS IN SECONDS/READ			
SEQUENTIAL READS			
1 sector	0.003	0.003	0.005
8 sectors	0.027	0.025	0.025
16 sectors	0.049	0.049	0.049
24 sectors	0.080	0.077	0.077
RANDOM READS, 1 SECTOR WIDTH OF SEEKS[b]			
0.10	0.039	0.034	0.044
0.33	0.044	0.051	0.055
0.50	0.059	0.059	0.059
0.90	0.070	0.076	0.076
RANDOM READS, 8 SECTORS WIDTH OF SEEKS[b]			
0.10	0.063	0.055	0.060
0.33	0.069	0.077	0.080
0.50	0.080	0.080	0.080
0.90	0.088	0.096	0.096

[a] *The sector total reflects the reduction caused by manual entry of bad clusters listed by CDC in the "flag track log" produced by the factory tests and supplied with the disk.*
[b] *Seek distance is the distance the heads traveled as a fraction of the width of the disk platter.*

As the *PC Tech Journal* fixed-disk benchmarks show, the CDC disk drives available from CORE International and Express Systems perform almost identically to the CMI drive that IBM offers for the AT.

TABLE 3: *Diskp Results*

	IBM	CORE	EXPRESS SYSTEMS
AVERAGE ACCESS TIMES (ms)			
Track-to-track	8.51	5.87	5.87
Random and track-to-track	23.56	21.80	21.87
Random	37.67	38.11	38.11

These benchmarks, developed by CORE International, measure average access time. The last number given (Random) is the one most often used by disk manufacturers to specify the "average access time" of their products.

data surfaces when power is lost or removed. Figure 1 shows the head-positioning system that is used in the drive. Special note should be made of the solenoid and sprocket ring. When power is taken away from the disk system, the solenoid releases, holding the sprocket in place and preventing any further head movement until power can be reapplied and the disk platters are spinning at the specified speed.

The only way to move the heads out of the normal data area before the power is turned off is to use the SHIPDISK.COM program supplied with the IBM diagnostics package or a program that does the same job. Of course, if the power fails, the heads just land where they are and may damage the disk surfaces if the computer is moved or jarred. IBM should instruct users always to park the heads using SHIPDISK before turning off the AT.

The CDC disk drive uses a rotary voice coil (RVC) positioner controlled by a closed-loop servo system in order to obtain accurate head positioning at high speed. A separate head and disk platter surface provide the necessary position feedback for the control loop. This method, which is extremely fast and significantly more expensive than competing band-stepper systems, has generally been used only in very high capacity disks for which the cost of the positioning system is just a small fraction of the total disk cost.

An automatic head-parking and locking feature is another advantage of the CDC disk drive. The disk heads are automatically moved out of the normal data area when power is turned off or fails, so sensitive data areas are not subject to damage. The solid mechanical design and sturdy construction of the CDC drive has already been noted. Just the look and feel of the drive are enough to inspire confidence. That first impression is reinforced by speedy, troublefree performance.

The CORE ATplus package can be purchased with an optional memory upgrade that brings a base-level AT (256KB) up to the enhanced level of 512KB. The ATplus disk drive is priced to match the cost of equivalent components sold by IBM. The documentation that CORE provides with its product consists of two photocopied pages that contain no fluff, but that include sufficient detail to describe the installation steps adequately. Most of the necessary instructions can be found in IBM's *Guide to Operations* for the AT.

The bad-block table for the CORE disk showed no bad sectors, but XENIX's media check program, **/etc/badtrack**, found one. PC-DOS, however, did not object to the offending sector, which fell in an unused part of the root directory area of the disk. An undetected bad sector in the file allocation or directory area under DOS could become a problem as the DOS disk fills with files over time. DOS needs to be improved considerably in its ability to detect and mark bad sectors.

EXPRESS COSTS LESS

Express Systems sells its drive at a lower price than CORE and delivers it already formatted for use with DOS 3.0. It also includes a helpful guide plus companion software to sweeten the deal. The guide covers a range of drives sold by Express Systems for the PC, XT, and AT. The amount of detail provided by Express Systems for the AT installation is comparable to that provided in CORE's documentation.

The Express Systems disk had a table of 12 bad tracks marked by CDC, but neither XENIX nor PC-DOS had any trouble reading or writing data to them. In practice, it is wise to enter the information manually about bad sectors

Technical Bulletin
No. 2 in a series.

SUBJECT: Engineering a LAN for Maximum Flexibility.

Quantum Software Systems Ltd. proudly announces QNX 2.0 — the Ultimate Distributed Network Operating System. QNX 2.0 is now available for the IBM-PC, IBM-AT, PC compatibles, DEC Rainbow and TANDY 2000. If you have been waiting for a Real-time Multi-tasking Multi-user Operating system with fourth generation LAN support, then QNX 2.0 can offer you today what the competition can't even begin to promise for the future.

QNX 2.0 integrates the Local Area Network architecture right into the heart of the operating system, at the fundamental level of intertask communication allowing tasks to communicate transparently with other tasks across the whole network. This means that any task (program/application) may access ANY serial port, ANY printer or ANY disk on the network. There are no artificial restricitons. Every PC with a disk is a potential file server. PCs without disks will automatically BOOT over the network.

QNX on the IBM-PC AT:

QNX is the first Multi-tasking Multi-user Operating system available for the AT. It is available in both networked and single machine configurations. At about 2.5 times faster than the QNX 8088 PC based systems, and 10 times faster than other multi-tasking operating systems on the same processor, QNX is the ideal program development environment.

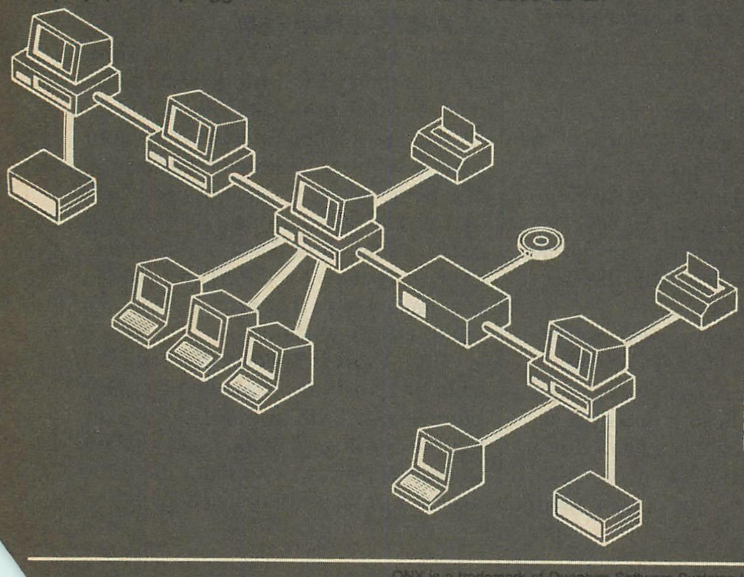
O/S	Computer	Processor	Measured time
QNX ™	IBM-PC AT	80286	480 usec
XENIX ™	Intel-286	80286	4,930 usec

File Security:

Designed with extensive file security features, QNX 2.0 provides login protection with network wide file permission checking based on 255 groups of 255 users. In addition, each PC user may control network access to devices attached locally to their machine.

Distributed Processing:

The QNX LAN supports distributed processing as well as distributed devices. Tasks may be executed on remote stations as easily as they may be executed on the local work station. This allows pure processing elements (PCs without keyboards or displays) to be plugged into the network to be used as an un-committed processing resource. This is ideal for real-time, process control, data acquisition and data communication applications.



Global Communications:

QNX supports a full implementation of X.25 allowing connection to public networks such as Telenet and Datapac. This allows you to link geographically separate LANs together providing true global area networking.

Cost Effective Growth and Flexible Solutions:

QNX is affordable, and will work with the PCs you use today and those you will use tomorrow. You may mix and match different brand PCs on the same QNX network with absolute ease. Multi-user expansion may be accomplished by adding terminals to PCs or PCs to the network. You can start your multi-user application on a single PC with 1 to 10 attached terminals. Once your single processor starts to show signs of degradation, add another PC and connect terminals to the new processor. If the disk becomes the major bottleneck, you may add hard disks to other attached PCs to distribute the processing. Applications which are very CPU intensive may wish to limit a single user to each processor and expand the system with low cost diskless PCs used as work stations. QNX does offer a truly cost effective and flexible solution to your applications needs.

Portability:

QNX 2.0 is portable. The operating system is independant of the physical local area network. It is available in a form suitable for porting to other 8088/8086/80186/80286 computers in the consumer, educational and industrial market place. QNX is ROMable and can operate in as little as 128Kb RAM.

DOS Compatibility:

PC-DOS version 2.1 can run as a task under the QNX 1.2 or 2.0 operating systems. QNX will also allow transparent access to the DOS file system partition and floppies.

QNX Products:

- QNX Operating System
- Full Screen Multi-terminal Editor
- Extended Utilities
- C Compiler & 8086 Assembler
- Basic Compiler
- Qbol (dibol) Compiler
- Text Processor
- Real Time Spelling Checker
- PC-DOS Emulator
- Electronic Mail
- Electronic Teleconferencing
- Full Screen Menu Developer
- Isam File Utility
- Networking Board
- OEM Customization Kit (to port QNX)

Established:

Quantum sold over 10,000 copies of its operating system during 1984, into all business systems environments, to developers of real time applications, government and educational systems, to software developers/integrators, universities and research establishments.

QUANTUM SOFTWARE SYSTEMS LTD.

Moodie Drive, HiTech Park, 215 Stafford Rd.
Ottawa, Canada K2H 9C1 (613) 726-1893

marked by the manufacturer, if the operating system provides a means to do so, as XENIX does (DOS does not, however).

The Express Systems software could be a real bonus because it includes a set of utilities for finding and marking bad sectors, which DOS has a bad habit of ignoring. Unfortunately, the BADBLOCK program marked almost 600 clusters as bad—that is, every 17th one, when in fact only a dozen clusters deserved such treatment. When Express Systems was notified of the bug, it supplied an updated version of the program with the error corrected.

The BBSET program, a utility that allows a user to update the file allocation table in order to mark bad tracks (such as those listed by the disk manufacturer) has an ambiguous user input system that makes it difficult to know how to type the input and what, if anything, is happening. Because each cluster consists of 4 sectors and there are 17 sectors per track, BBSET marks 5 clusters as unusable in the FAT for each track noted as bad.

Although the few bugs that were discovered while testing these Express Systems programs do not inspire confidence in their reliability, the company seems to be working diligently to correct the problems, and the programs nonetheless do help in overcoming DOS's deficiency in locating and marking bad sectors.

ATplus 20MB disk: $1,595
Memory (256KB): $495
CORE International, Inc.
542 S.E. 5th Avenue
Delray Beach, Florida 33444
305/276-3929
CIRCLE 302 ON READER SERVICE CARD

PC/AT Fixed-disk Drive Option: $1,595
IBM
Entry Level Systems
P.O. Box 1328
Boca Raton, FL 33433
305/982-4700
CIRCLE 304 ON READER SERVICE CARD

20MB disk: $1,340
Express Systems, Inc.
1254 1/2 Remington Road
Schaumberg, IL 60195
Within Illinois: 312/882-7733, ext. 200
Outside Illinois: 800/341-7549, ext. 200
CIRCLE 305 ON READER SERVICE CARD

Augie Hansen, a contributing editor to this magazine, owns Omniware, a software development and training company based in Denver, Colorado. He is currently writing a series of books about UNIX.

FIGURE 1: *CMI Head Positioning System*

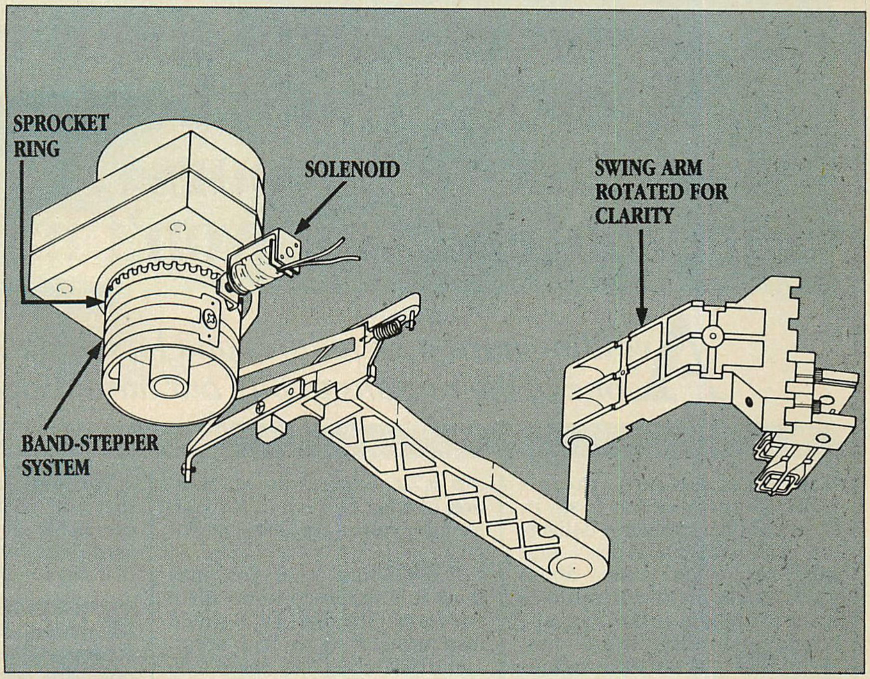


ILLUSTRATION • COURTESY OF COMPUTER MEMORIES INCORPORATED.

The CMI drive is designed so that when power is lost, the solenoid releases, holding the sprocket ring in place and leaving the heads over the data area. Damage to the data may therefore occur if the disk is moved without first parking the heads.

SUSPICIOUS SPECIFICATIONS

Product announcements and advertisements for replacement drives for the PC/AT have to be viewed with skepticism. The way some claims are stated makes it difficult or impossible to know what is real and what isn't. Many of the replacement drives being offered are said to be "AT-compatible," but what does that really mean?

Most of the suppliers say that their drive could be plugged into the AT and could be identified as one of the supported types. Some say that their drives are type 2, like the CMI drive, but that they are slower than the CMI drive by a factor of two or more.

Actual performance degradation depends on the interleave factor and the user's mix of processor tasks and disk I/O. This is true of some half-height 20MB drives that have an average access time of 80 ms.

IBM advertises the AT as supporting a maximum of three drives, either two diskettes and one fixed disk or two fixed disks and one diskette. This is actually a space limitation and not due to the Western Digital controller, which can control four devices. The advantage of the half-height drives is that with the proper bracketing, it would be possible to install two fixed disks and two floppies for a total of four drives.

Several drives of larger capacity have more than the requisite access speed, but because of the larger storage capacities, they must be identified as something other than type 2 in the AT's configuration table. Some programs do not deal with this effectively. PC-DOS itself must be tricked into using storage beyond 32MB, and XENIX must be patched if it is to use any drive other than the standard type 2. Many of the replacement drives must have accompanying program patches, device drivers, or revised BIOS ROMs to run with some programs.

This is not to say that disk drives of larger sizes are to be completely avoided at all costs. They certainly serve a purpose, and in most cases will be very satisfactory solutions to user problems, such as managing a large database and providing sufficient user file space in multiuser situations. However, where true AT compatibility is required, such drives will not do the job.

—*AH*

If you believe these simple facts about hard disks, you'd be willing to pay more for ours.... fortunately, you won't have to

Avoiding hard disk failures and loss of data is just one of the reasons to buy our hard disks.

Hard disks can fail—there's really no other nice way to say it. Even IBM has problems delivering ATs with hard disks that work. We're not talking about nice, clean, clear-cut failures where the drive seizes up, coughs, and rolls over and dies. We're talking about the insidious little creeping failures that sneak up over time—like a missing sector here or a lost sub-directory there.

There are precautions you can take to protect against failure and ultimate loss of data. Here is what we contribute toward minimizing the potential loss of your data.

Best Drives Available

First, we buy the best drives available. Sounds trite, doesn't it? I mean, a drive's a drive—right? Hardly. You should see some of the junk we get in our labs. Some have such high failure rates that we even questioned our own $10,000 hard disk tester. But when we tested other manufacturers' drives we were assured that our equipment was fine, which just confirmed that the bad hard disks were not only bad—they were real bad.

But that's just the weeding out process. We then take each drive that we've put through our tester and test it again with the controller you've requested. We call this a "tested pair."

DOS Doesn't Do It

In case you're thinking that all this is an unnecessary duplication of what DOS does for you, let me explain the basic facts of life.

If DOS did what you may think it is supposed to do when you format the disk, DOS would map around these bad areas. Unfortunately, DOS doesn't do this.

DOS 2.0 and 2.1 can't enter the bad tracks. DOS 3.0 can, but only on the IBM AT. Unfortunately, as the press has so well documentated, the AT's hard disk develops bad tracks later on.

We do what DOS can't

We believe the problem is so bad, we use a software program that performs a powerful test of your disk drive on all of the IBM or IBM compatible computers—PCs, XTs, and ATs. Our format takes hours to analyze the disk. But when we finish, you know that the bad tracks are really mapped out so you won't write good data that will disappear into a black hole. We even send you a printed statement of our test results.

Our software allows you to type in the bad track locations from the list supplied by the manufacturers, so you'll never write good data to them—even if DOS didn't identify them as bad. The software even lets you save the location of these bad sections to a file, so that you can reformat your disk without spending hours retesting.

We even include a program that will give you continuous comments on the status of your hard disk. No more waiting for that catastrophic failure.

Average Access Time

As you might suspect, some hard disks are faster than others in their ability to move from one track of data to another. The time it takes the hard disk to move one-half way between the beginning of the disk to the end is called the "average access time."

The first generation of 10 megabyte hard disks had average access times of 80-85 milliseconds (msec). But computer users love speed, and guess what—the average access time for the new 20 megabyte hard disk in the IBM AT is only 40 msec. (We sell an AT equivalent with only 30 msec access time!)

There are some legitimate reasons for the shorter access time. It's particularly helpful when there are multiple users on the same hard disk. It's also important when running a compiler. But remember, before you get too wrapped up in the access speed, there's always that ST 506 interface which won't let data transfer from the hard disk to the computer any faster than 5 megabits/second. We've bypassed that choke hole, too. If you want the functional equivalent of a Ferrari with a turbocharger, order our 10 Mbit per second 108 megabyte hard disk with 18 msec of average access speed.

Compatibility

To be sure that your hard disk is 100 percent compatible with the IBM XT you don't need to buy the same hard disk that's in the XT. You can't even be sure what brand hard disk it is because IBM, like Express Systems, goes into the marketplace and buys hard disks from several vendors. However, they buy their XT hard disk controller from only one vendor—the same one we do.

You can buy the IBM XT controller from IBM for $495 or you can buy from us, the functional equivalent, manufactured by the same company that makes it for IBM for only $195. Is it the exactly identical IBM XT controller? No, it's better. First, it takes less power, and secondly, it can control from 5 to 32 megabytes—the IBM controller can work with only 10 megabytes. It is 100 percent IBM XT compatible, and 100 percent is 100 percent. If you want to save a slot, we carry a version that lets you operate two hard disks and two floppy disk drives.

More than 32 Megabytes

You can operate with more than 32 megabytes (the limit of DOS) through the use of "device drivers." Express Systems can supply you with device drivers for our hard disks for over 32 megabytes formatted. But, if you don't have individual files, or databases that are large, you might want to consider one of our controllers that can divide our 65 megabyate (formatted) hard disk into two equal volumes of 32 megabytes each.

Reliability

We offer you a choice between iron oxide and plated media—the stuff that covers the hard disk and gives it its magnetic properties. Iron oxide is,—well, it's rust. If you inadvertently joust your disk, you may cause the low flying head to dig out some iron oxide. A little rust flake can ruin your whole day. Plated media is more resistant to damage, and if it happens, less data is lost.

We offer both types of hard disks. The iron oxide is older

technology, and quite frankly, manufacturers understand it better. Their better understanding, combined with some of the special head locking mechanisms, gives us peace of mind when we sell you one.

Power

Hard disks consume power. Our small, half-high hard disks consume so little power that you can use them with your existing IBM PC power supply. If you plan to use lots of slots, you'll want to increase your power supply to be safe. We offer the same amount of power for your PC that comes in the XT.

Our Customers

Some folks just never feel comfortable buying mail order. They forget that Sears began as a mail order house or that IBM is now into mail order. But, if it helps, here is a *partial* list of customers who have felt comfortable to buy from us.

IBM	Sears
American Express	Honeywell
U.S. Army	MIT
AT&T (Bell Labs)	RCA
Bausch & Lomb	Lockheed
Xerox	Sperry

Easy to Install

If you're like most of us, raised on the boob tube rather than the Great Books, you'd rather see the movie than read the book. Well, now you can choose to read our installation manual or for only $9.95 more, you can get a VHS or Beta video cassette showing the simple steps for installation.

Our VHS or Beta Cassettes make installation easy.

Warranty

We offer you a one year warranty on our hard disks—the same as IBM on the AT and 90 days on the tape drives. (It's all the manufacturer gives us.) If anything goes wrong with your tape or disk drive or hard disk, send it back in the box it came in. However, we have found that we can usually solve the problem over the phone. So call first for a return authorization number because we can't accept any returns without it.




Complete Hard Disk Kits

Formatted Storage Capacity in Mbytes	Height	Plated Media	Average Access	Transfer Rate	PC or PC/XT	AT
10	½	yes	85 msec	5 Mbits/s	$ 625	$ 430
10	Full	yes	85 msec	5 Mbits/s	$ 625	$ 430
21	½	yes	85 msec	5 Mbits/s	$ 825	$ 630
21	Full	no	30 msec	5 Mbits/s	$ 1,535	$ 1,340
32	½	yes	85 msec	5 Mbits/s	$ 1,095	$ 895
32	Full	no	30 msec	5 Mbits/s	$ 1,775	$ 1,575
65	Full	no	30 msec	5 Mbits/s	$ 2,295	$ 2,070
108	Full	yes	18 msec	10 Mbits/s	$ 4,995	$ 4,995
Removable Hard Disk						
10	½	no	90 msec	5 Mbits/s	$ 1,095	N/A

Tape Systems and Subsystems

Formatted Storage Capacity	Height	Data Transfer Rate (k/sec)	PC or PC/XT	AT
60 Mybytes	½	88	$ 995	$ 995
60 Mbytes Subsystem		88	$ 1,295	$ 1,295
17.6 Mbytes Start/stop Subsystem		24	$ 795	$ 795

Controllers
All of our hard disk and tape controllers are available separately: Please call for prices.

Subsystem Chassis
Any of our disk or tape units are available in an external subsystem for an additional $250.00. You can mix & match any of our ½ high hard disks or tape drives together or add any single full height hard disk.

Tape Cartridges $ 35.00
Express Certified 555 foot 310 Hci ¼-inch Data Cartridge

Power Supply $ 99.50*
130 Watt Power supply
*with the purchase of any drive

Comes complete

All Express Systems products come complete with the appropriate software, tape and/or hard disk controllers, and cables where required. Hard disks are formatted and tested with the PC DOS of your choice. All drive sizes are formatted capacities.

If your application requires a stacking kit, power splitter cables, daisy chain cable, or some other variation, we'll supply these items at a nominal charge. We even ship our hard disks with Command Assist™ an on-line DOS-like manual to give you help with your DOS commands.

More questions?

Because we spend so much attention on the front end with ensuring that our disks will arrive in working order, we have a customer service department that, unlike many of our competitors, has little to do. When you need us, you won't get a constant busy signal.

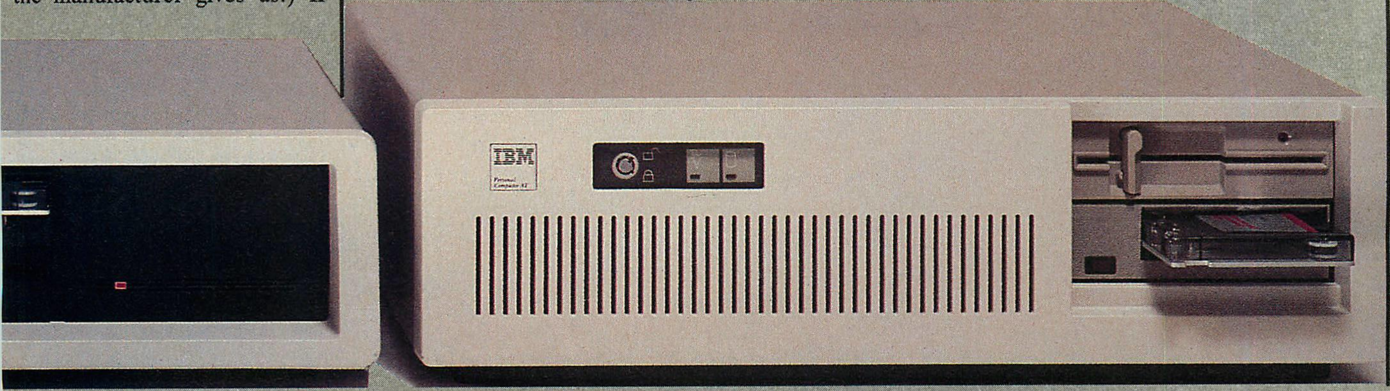
Call our friendly, knowledgable customer service staff to get answers to your questions—before or after the sale. Our people, who know the PC, can talk you through the sticky parts, and they'll respond to you quickly. Just call us.

How to order

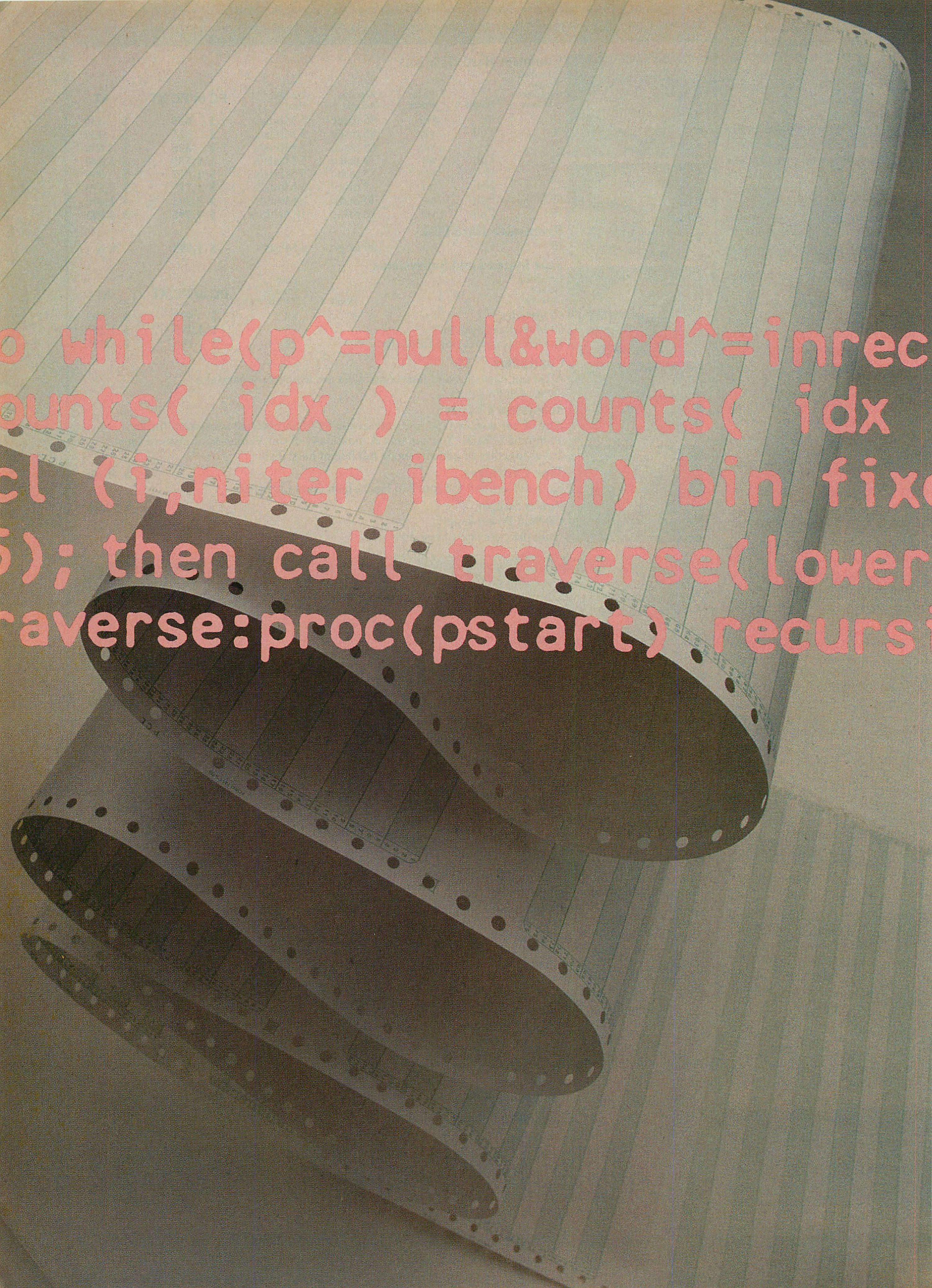
Pick up the telephone and call 1-800-341-7549, to order. We accept Master Card, VISA, American Express and Diners Club. Or send a cashier's check or money order (We'll take a check, but you'll have to wait for it to clear) and tell us if you want one of our recommended configurations or you want to mix and match yourself. Corporations with a DUNS number may send purchase orders for quantities over five.

EXPRESS SYSTEMS

Call Toll Free 1-800-341-7549 Ext. 900
In Illinois call (312) 882-7733 Ext. 900
Express Systems, Inc., 1254 Remington, Schaumburg, IL 60195
CIRCLE NO. 143 ON READER SERVICE CARD



Command Assist is a trademark of Micro Design International


o while(p^=null&word^=inrec
ounts(idx) = counts(idx
cl (i,niter,ibench) bin fix
5); then call traverse(lower
raverse:proc(pstart) recurs

Considering PL/1

An overall successful first porting to the PC for this relatively untapped language

ROBERT BARNES

Many programmers think of PL/1 as an IBM-sponsored aberration that pleases no one because it tries to please everyone. For these persons, the release of the PL/1 compiler for the PC by Digital Research, Inc. is about as interesting as a new operating system for IBM System/360s. After all, if the U.S. Department of Defense rejected PL/1 and spent millions of dollars designing a new language (ADA) based on Pascal, there must be something wrong with PL/1. For anyone who believes that the DOD spends its money wisely and that its requirements are the same as his, read no further: Pascal and its derivatives must be better. For everyone else, however, PL/1 deserves more serious consideration.

PL/1 is a powerful programming language and its application on the IBM PC should raise the attention of two sets of users: those who need a language with powerful facilities, either as their first choice or as a replacement to a language they have found limiting, and those who are already familiar with PL/1, probably on mainframes, and are looking to apply similar skills and techniques on the PC. This article reviews the language itself and the compiler that ports it to the PC, addressing questions that might be raised by either group of potential users.

When PL/1 was first announced in the mid-sixties, it was seen as an answer to the deficiencies of COBOL and FORTRAN. PL/1 was promoted (and named) as "one language for commercial, scientific, and systems programming" and was said to combine the test features of a number of languages. Commercial programmers could define files and record structures and manipulate them much as in COBOL. For mathematical and engineering programming, PL/1 offered all the facilities of FORTRAN.

ALGOL influence, although not admitted publicly, was clearly strong: PL/1

ANSI defined a subset language, Subset G (for general use), that omitted many esoteric areas of complete PL/1.

used the structured logic of IF.. THEN [ELSE] and DO ... END rather than the unstructured logic forms still used in COBOL and FORTRAN, and it used ALGOL's concept of procedure blocks, providing the potential for locally-scoped variables and recursion. PL/1 also included list-processing facilities (the ability to create pointer-linked information structures in memory) and text manipulation features that previously were provided only by specialized languages or low-level routines. It was one of the first languages designed to take advantage of interrupts, a concept that developed with the introduction of operating systems.

Yet in spite of all this, PL/1 never really caught on. COBOL remained the dominant commercial programming language, while scientific users gradually moved away from FORTRAN, and then usually not to PL/1. This was partly a matter of timing—by the time that good PL/1 compilers became available, most IBM users had already made the switch from low-level programming to COBOL. Although PL/1 was better, it was not so much better as to justify retraining costs, and with COBOL-trained programmers abundant (at least compared to PL/1) the COBOL preference became self-perpetuating.

Other problems involved PL/1's intrinsic complexity and flexibility: PL/1 allowed programmers to write COBOL-like, or FORTRAN-like, or ALGOL-like programs. Yet PL/1 was nevertheless different from those other languages, and without today's understanding of structured programming concepts (which PL/1 supported, but did not enforce) and hindered by IBM's early compilers, programmers of the day often decided that PL/1 was unnecessarily complex. This, in combination with its overall rejection by the commercial world and the fact that instructors preferred to teach a smaller language, such as Pascal

PHOTOGRAPH • CARL FISCHER

(which enforced structured programming), dashed PL/1's potential.

The sheer size of PL/1 also is a problem. Few users understand all of the facilities of the full language and it is a language that is difficult to implement, although its most complex features are rarely used and easily omitted. ANSI therefore defined a subset language, Subset G (for general use), that omitted many of the more esoteric areas of the complete PL/1, areas such as multitasking support and macro (preprocessor) facilities. Subset G contained all of PL/1's essential features but was implemented more easily on minicomputer and microcomputer systems. In fact, the omissions and the greater stringency of some rules actually may have resulted in a better language.

TABLE 1: *Compilation Speed and Program Size*

PROGRAM	SOURCE CODE	COMPILE TIME PC	COMPILE TIME PC/XT	LINK TIME PC	LINK TIME PC/XT	.OBJ SIZE (bytes)	.EXE SIZE (bytes)
HELLO.PLI (listing 1)	3 lines (69 bytes)	22	9	53	27	384	13,312
KWIC.PLI (listing 2)	124 lines (2,944 bytes)	64	34	61	34	3,584	17,920
PENTATH.PLI (listing 4)	132 lines (4,096 bytes)	57	33	59	36	3,456	20,480

Compile and link times are in seconds.

DRI PL/1 is a three-pass compiler that produces a .OBJ file from program source code that is prepared with any text editor. The .OBJ file is then linked with any other required files and the PL/1 runtime support library to produce a .EXE file.

TABLE 2: *Compilation of Larger Programs*

REPLICATION FACTOR	SOURCE CODE (bytes)	.OBJ SIZE (bytes)	COMPILE TIME (seconds)
1	3,456	3,584	45.4
2	6,912	6,144	70.7
3	10,368	8,576	90.4
4	13,824	11,008	125.7
8	27,392	20,992	229
16	51,784	40,960	445.2

Link times are not listed because they were relatively constant; the largest program linked in 111 seconds to a .EXE size of about 46,000 bytes.

DRI PL/1

The Digital Research, Inc. PL/1-86 1.0 compiler for the IBM PC is a PC-DOS adaptation of DRI's earlier PL/1-80 compiler, available for CP/M systems for some time. It uses DOS 1.1 or later; DOS 2.0 was used in this evaluation (DOS 3.0 was used on the PC/XT).

The system comes with two PL/1 manuals, a language reference manual and a programmer's guide; a programmer's utility guide; and four floppy disks containing the software. The compiler is contained comfortably on one disk, and much of a second and all of a third disk contain sample programs. The fourth disk is a general DRI programmer's utility disk, which is packaged with all DRI compilers. In addition to the linkage editor, which, combined with the PL/1 runtime subroutine library on another disk, is necessary to prepare compiled PL/1 programs for execution, the utility disk contains a relocating 8086 assembler, a software librarian, and a cross reference utility.

For the user who knows enough about DOS to work around errors in the instructions, installation is easy. The disk contents are described in the READ.ME file on the disk itself and refer, for example, to a file called PLI.CMD when in fact the file is PLI.EXE. In another place, the user is instructed to invoke the linkage editor with the command LINK, but there was no LINK.EXE file supplied, and linking with the IBM-supplied LINK was not successful. Instead, the command in the documentation should have been LINK86, the file supplied for DOS users.

Even though potential users of a language compiler can be presumed to be more computer literate than the target market for a spreadsheet, the installation instructions should make the compiler as simple to install as Lotus. Although the manuals contain most of the information necessary to installation, the location of that information in the sixth chapter of the second manual is odd. (To be sure, the foreword and table of contents steer the user to this chapter for using the system.)

The system used was configured by copying the PL/1 compiler to a system disk containing a text editor and copying all of the second and fourth disks to a second disk. This provided sufficient space to compile small programs (a couple hundred lines) from the compiler disk, leaving the linkage editor and PL/1 subroutine library in drive B:, so that all stages from writing to testing could be accomplished without disk changes. Larger programs required the compiler and the linkage editor to be swapped through the default drive (which is not necessarily A:) while the program was written on drive B:. DRI designed the software to be usable without excessive disk changes; a .BAT file can be created to control compiling, linking, and testing.

DRI PL/1 is a three-pass compiler that produces a .OBJ file from program source code that is prepared with any text editor. The .OBJ file is then linked with any other required files (such as those from separately compiled routines) and with the PL/1 runtime support library to produce a .EXE file. DRI supplies a linkage editor. The resulting .EXE file requires no other runtime support in order to execute.

Table 1 lists compile times for three programs using a PC with two floppy-disk drives and a PC/XT with the software configuration described above. Listing 1 (HELLO.PLI), a very short program, simply prints hello. Listing 2 (KWIC.PLI) extracts words from text to form an alphabetized list of all words that occur in the text (it calls listing 3, WORDSTR.PLI). Listing 4 (PENTATH.PLI) is a PL/1 version of the pentathlon benchmark (see "C and the PC, Part 1," William J. Hunt, *PC Tech Journal*, November/December 1983, p. 110). Timings were taken with a stopwatch. Compile time is sensitive to compiler options; in this case, the option to list the program was used.

In order to test the compiler on larger programs, KWIC.PL1 (listing 2) was multiplied in size by copying the program into itself a number of times with adjustments so that it would compile cleanly. The programs were compiled using a .BAT file that displayed the system time before and after the PL/1 compilation to obtain a more accurate time than with a stopwatch. Compiler options were reduced to the defaults. Table 2 lists the results of this test. Link times are not listed because they were relatively constant; the largest program linked in 111 seconds to a .EXE size of about 46,000 bytes. Compiling and linking is solidly I/O bound, and times improve when a fixed disk is used. Another test of the compiler's performance with larger programs, an attempt to compile a program written for full PL/1, is discussed later. (DRI PL/1 does not claim this ability.)

LANGUAGE LEVEL

The DRI PL/1 compiler is designed to support Subset G PL/1, but it does not provide some features and includes some extensions. In fact, what is standard and what is an extension to Subset G is sometimes confused. Some features that are implemented as Subset G standards by DRI are considered extensions by some other (albeit mainframe) PL/1 compiler producers.

The reference manual documents the differences between DRI PL/1 and the standard. A more helpful comparison might list the differences between this PL/1 and IBM's full PL/1. The main features of PL/1 are summarized below.

Language level 1—program structure. A PL/1 program consists of one or more external (separately compiled) procedures, each of which may contain internal Procedure and Begin blocks. Blocks may contain (local) variables and may be nested within other blocks.

Procedures (internal or external) can pass and return arguments, using either call by reference or call by value. A (documented) fault of the present release is that constants are passed by reference, potentially causing errors if the corresponding argument is changed in the subprogram. Procedures can be written as functions and can be recursive. Like Subset G, DRI PL/1 does not support the REENTRANT option (which allows a single copy of a procedure to be used by two or more tasks at once).

Language level 2—data types. DRI's PL/1 supports the full range of problem data types: DECIMAL FIXED, BINARY FIXED, and FLOAT for numeric data and CHARACTER and BIT strings. PL/1 has no equivalent of the Pascal TYPE statement that permits programmers to define their own data types.

As in full PL/1, a precisely defined set of conversion rules allows automatic conversion, where valid for the actual data, between any type, even when one of the variables is not a numeric type. This is helpful, for example, in situations where a character field contains a number that is used in arithmetic:

```
DCL FIELD CHARACTER (6);
DCL NBR DECIMAL FIXED (6);
   . . .
IF VERIFY (FIELD, '0123456789')=0
/*check to see if numeric*/
THEN NBR=FIELD;
```

But this flexibility can be counterproductive, and warning messages would help. For example, a Boolean (BIT (1)) variable can be set with

```
VALID=BALANCE > LIMIT;
```

but PL/1's conversion flexibility allows the comparison result to be assigned to any data type, with appropriate conversions. I have more often mispunched a dash as an equal sign, erroneously giving A=B=C, than I have deliberately assigned 1 or 0 to a number by testing a condition with this code. A good case could be made for implementing a warning expression whenever a conversion between string and arithmetic occurred and whenever a conditional expression was assigned to anything other than a BIT(1) field. (Full PL/1 does not issue this warning either.)

The range of precisions available is restricted by the PC architecture. The maximum precision for a binary number, for example, is 15 and the default is 7; the mainframe maximum is 31 and the default is 15.

DRI PL/1 does not support the PICTURE attribute, although it does provide the PICTURE format item for STREAM I/O. DRI PL/1 supports data of the forms STATIC, AUTOMATIC (data are in implicit procedure stack), BASED (address indeterminate, given by POINTER variable), and EXTERNAL (known to all procedures linked into the program), but does not support data of the type DEFINED (same address as another variable). Like Subset G, it does not support CONTROLLED (explicit stack) data. However, both can be simulated with BASED data. For example, DEFINED may be simulated as follows:

```
DCL A CHAR(100),
        B CHAR(10) BASED (P),
        P PTR;
P = ADDR (A);
```

A simple DEFINED attribute would be convenient and less likely to produce errors, but its full PL/1 subtleties might not be missed.

Data may be initialized with the INITIAL attribute only if it is STATIC. Scalar data may be combined IN arrays and structures. DRI PL/1 does not support the use of arrays in expressions or assignments; the user cannot write

```
DCL A(10) BIN FIXED;
A = 0;
/* set all elements of A to zero */
```

This is merely an inconvenience, because he can write

```
DCL I BIN FIXED;
DO I = I TO DIM(A,1);
   /* DIM(A,1) = 10 in this case*/
   A(I)=0;
END;
```

Language level 3—logic control. For conditional logic, PL/1 uses the IF statement:

```
IF condition
THEN one or more statements
[ELSE one or more statements]
```

For loops, PL/1 uses DO:

```
DO I = 1 TO 20;
   /* statements will be done */
   /* twenty times */
END;
DO WHILE(X<Y);
   /* will be done for as long */
   /* as X is less than Y*/
END;
```

Logic may be nested without limit.

DRI PL/1 implements IF and DO to the full extent of the Subset G definition, but this is not enough. PL/1 initially contained no Case facility and instead relied on LABEL variables and GO TO statements to implement multiple choice logic:

```
   DCL CASE(10) LABEL;
   GO TO CASE(TYPE);
CASE(1): ...
   GO TO END_CASE;
CASE(2): ...
   GO TO END_CASE;
   ...
```

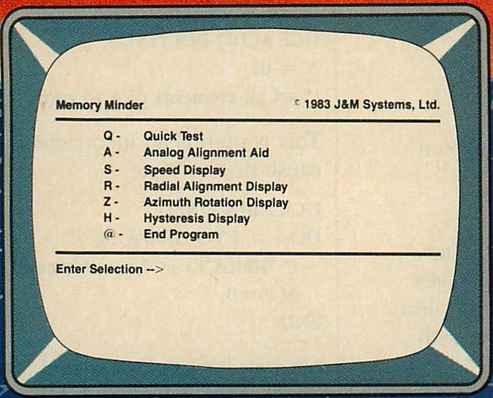
Not very elegant. Once structured programming became commonplace, IBM added the SELECT statement to full PL/1 to provide a better implementation:

```
SELECT (TYPE);
   WHEN(1)...
   WHEN(2)...
   OTHERWISE...
END;
```

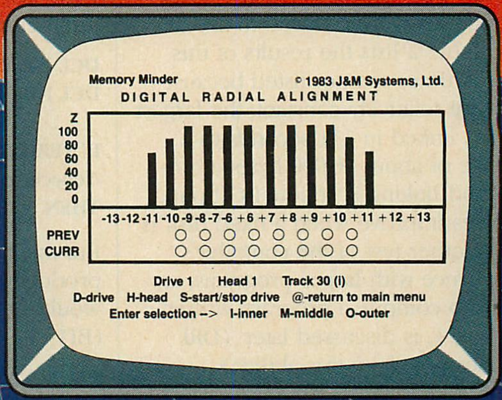
SELECT is probably the single most important facility added to PL/1 in the last

MEMORY MINDER T.M.

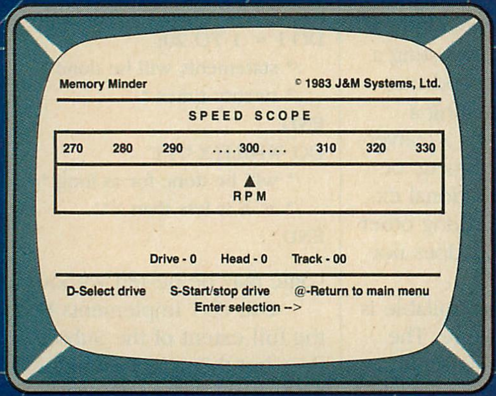
. . . A UNIQUE APPROACH TO DISK RELIABILITY!



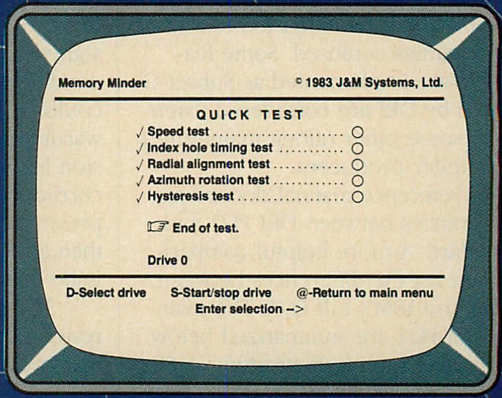
Select any one of seven tests to perform preventive maintenance or to isolate problems. Simple, single-letter commands make MM easy to use! Use MM to align the head, or adjust the speed.



Use the MM Radial Alignment Test to check the head alignment of your drives. No need for an oscilloscope or other expensive test equipment!



Check the motor speed of your drives. Or, you can even use the Speed Test to adjust the drive speed. No need for any test equipment!



Use the Quick Test to quickly and automatically test five of the most important performance parameters of your drive. Monitor your drives for long term drift. Isolate problems quickly and automatically!

PROTECT YOUR DATA.

Now you can make sure your data is being recorded properly by the use of the revolutionary ***Memory Minder.***

The Memory Minder from J & M Systems, tests your disk's performance and calibration without any additional equipment! It measures your disk's performance and displays it on your screen.

This is the most comprehensive disk diagnostic program available for your IBM microcomputer. You can even adjust drive alignment while watching the display!

Spot problems ***before*** they endanger your data! If you own a disk drive, you ***need*** the ***Memory Minder!***

IBM	Price
1 - 160K Drives	$79
2 - 320K Drives (Also test 160K drives)	$99

Memory Minder also available for most IBM compatibles.

J & M SYSTEMS IS THE DRIVING FORCE!

J & M SYSTEMS, LTD.
15100-A CENTRAL SE
ALBUQUERQUE, NEW MEXICO 87123
505/292-4182

CIRCLE NO. 192 ON READER SERVICE CARD

15 years. Unfortunately, this was not used in ANSI standard PL/1 and its absence significantly detracts from PL/1 usability. DRI should not wait for ANSI to move before implementing it.

Language level 4—interrupt handling. PL/1 was one of the first languages that was designed for environments with operating systems that could raise interrupts. In PL/1 the ON statement specifies that "ON the occurrence of a particular condition" some task will be performed. For example:

```
ON ERROR
  /*an error condition has been */
  /* detected by the operating system */
BEGIN;
  PUT...
  /*print various diagnostics */
END;
```

Definition of ON conditions is somewhat system-dependent; the original IBM definition clearly reflects System/360 architecture. Table 3 lists the error-trapping statements provided by the DRI compiler. This is less than half the list available in full PL/1, but the omissions are features such as GET DATA that either are not present in the subset or are caught by other ON conditions. The DRI implementation of error-condition subcodes is an attractive feature.

Subset G PL/1 and the DRI compiler omit the full PL/1 concept of condition prefixes, thereby removing a poorly understood and largely unused feature of that language. The compiler is, however, left without the diagnostic facilities that are provided by SUBSCRIPTRANGE and CHECK.

Language level 5—I/O facilities. DRI PL/1 provides a device-independent I/O system that allows data transmission between memory and console, printer, and disk files. Both direct access and sequential files may be processed.

PL/1 provides two kinds of I/O: STREAM I/O and RECORD I/O. STREAM I/O emphasizes ease of use and is oriented toward sequential reading and writing of character format data. For example, the following

```
DCL A DEC FIXED (3),
    B CHAR(10),
    C BIN FIXED;
GET SKIP EDIT(B, A, C) (A(10), F(5),
  F(5));
```

interprets the first 10 characters of the line as the value for B, the next 5 as A's value, and the next 5 as C's value. The characters in B's and C's positions are converted to the relevant types for storage. LIST is an alternative to EDIT that reads (GET) and writes (PUT) data in

TABLE 3: *Error Messages*

GENERAL ERROR TRAPPING
ON ERROR
ARITHMETIC ERROR CONDITIONS
ON FIXEDOVERFLOW
ON OVERFLOW
ON UNDERFLOW
ON ZERODIVIDE
I/O CONDITIONS
ON ENDFILE (end of file)
ON ENDPAGE (end of page detected)
ON KEY (invalid key)
ON UNDEFINEDFILE (file cannot be opened)

The DRI compiler offers fewer than half the error-trapping statements available in full PL/1. The DRI implementation of error-condition subcodes is an attractive feature.

free format. Depending on the options in the file's OPEN statement, items may be separated by commas and character items may be enclosed in quotes.

RECORD I/O is used to read and write data to files without conversion, providing both sequential and direct access to stored files. Figure 1 shows an example of such I/O.

It would be helpful to see a discussion of the differences between file handling on a PC and file handling on a mainframe. Indeed, some programmers already familiar with PL/1 on mainframes may have more difficulty with the differences between DOS and VM (such as the presence of line feed characters in the input) than they will with PL/1 dialect differences.

Language level 6—preprocessor. DRI PL/1 has %INCLUDE to copy external code into a program (typically used for record definitions) and defines %REPLACE as the basic text replacement facility instead of the more flexible, but complicated, preprocessor facilities of full PL/1. Not only does this simplify implementation and make it possible without a special preprocessor compiler pass, it provides the only preprocessor feature that most PL/1 programmers understand and use.

Language level 7—built-in functions. Included in DRI's wide range of built-in functions are most of the (relevant) functions of full PL/1. The only omissions from the Subset G standard are ATANH, STRING, and VALID, and ASCII and its inverse, RANK, have been added. ATANH is somewhat obscure (an inverse hyperbolic tangent) and can be derived from other included functions. STRING can be simulated by redefinition, and VALID can be simulated with VERIFY (at least in the absence of a PICTURE attribute).

Additions to full PL/1 are COLLATE, COPY, TRIM, and REVERSE. The manual also defines LOCK and UNLOCK, although the fine print reveals that these do not apply to the PC-DOS version (which supports only single-user processing). Omissions are ANY, ALL, CURRENT STORAGE, LOW, HIGH, OFFSET, ONCHAR, ONCOUNT, ONLOC, ONSOURCE, POINTER, POLY, PRECISION, PROD, REAL, REPEAT, SAMEKEY, SIGN, STATUS, STORAGE, and SUM. Inexplicably, TIME is regarded as "not supported by the operating system."

BEYOND PL/1

In PL/1, subroutine calls and function references, with or without arguments, may be used to invoke subprograms. If the subprograms named are neither internal procedures nor built-in functions, they are defined as ENTRY names in DECLARE statements, and refer to externally compiled subprograms. Such subprograms need not be written in PL/1, but may be written in assembly language or another high-level language.

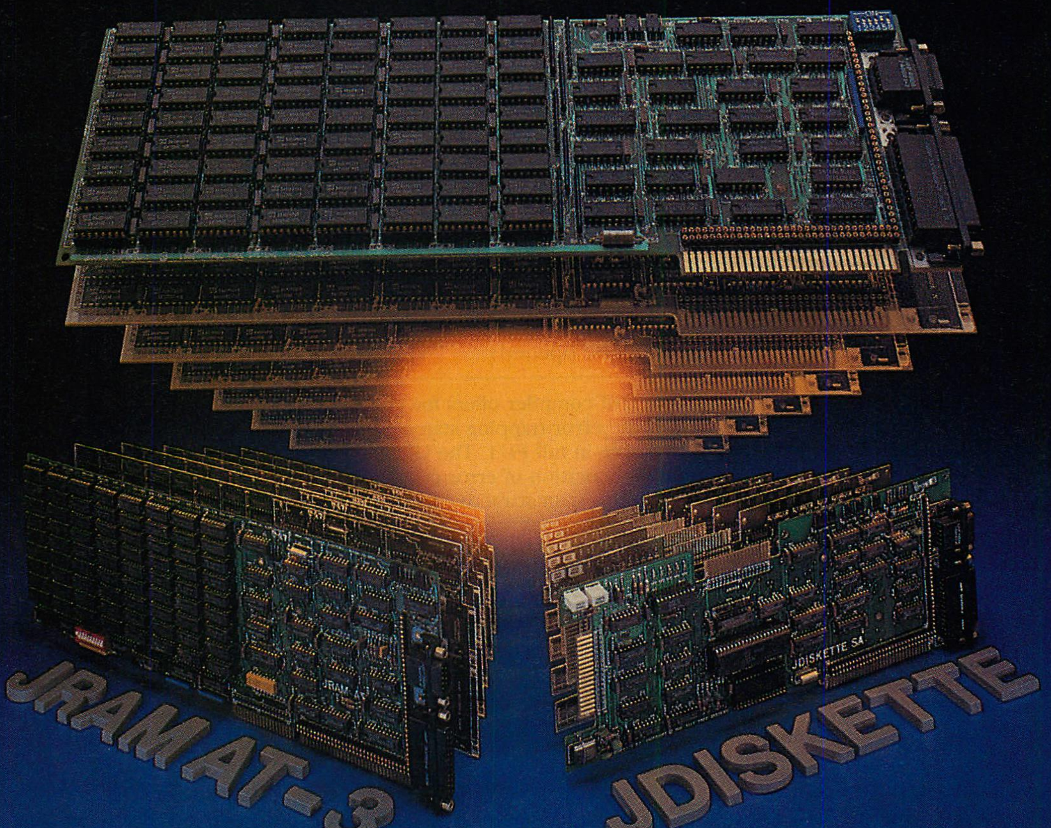
The DRI PL/1 language reference manual includes two appendices: one describes the data formats used by each type of variable, while the other details the interface conventions used by PL/1. Included in the installation disks are sample programs showing direct calls to each operating system function and assembly language source code for direct operating system calls.

With the relocating assembler, link editor, and cross reference utility that are included, the support for extended programmer functions is excellent. In addition, optional subroutines (contained in assembly language programs, in source code) provide direct access to all of the operating system functions. Sample PL/1 programs using these programs are included on the disks. Writing PL/1 programs to use function calls is a simple matter of finding the appropriate example and using it. This simplicity is misleading, however, because the subroutines are in *DRI assembler* source code form. No information is given on linking DRI PL/1 programs with programs written in languages other than PL/1 or (DRI) assembler. The standard IBM LINK does not work, so programs presumably are compatible only with other DRI compilers.

No mention is made in either of the PL/1 manuals about a facility for producing sound or graphics; but the

THE FAMILY TREE

JRAM-3



BRANCHING OUT

Tall Tree Systems presents JRAM-3, the newest member of the JRAM family. JRAM-3 is a fourth generation multifunction memory board and the successor of the highly praised JRAM-2. Designed to meet the latest expanded memory specification standard being implemented by the major spreadsheet vendors, JRAM-3 can access up to eight megabytes of memory for larger, more efficient spreadsheets. JRAM-3 can also be used for DOS memory, electronic disk, print spooler, and program swapping applications!

Determined to maintain our reputation as the price leader in memory expansion, Tall Tree Systems offers JRAM-3 fully populated with two megabytes for an amazing $699. A JETDRIVE/JSPOOL combo disk is included free of charge. This is the same highly acclaimed JRAM software that has helped make Tall Tree Systems the pioneer in the industry for bank switched memory and RAM disk technology. The new combo disk features JPAGER which allows any expanded memory application program to utilize multi-megabyte memory.

The family tree wouldn't be complete without an AT version. Look to us in September for the new JRAM AT-3, our third generation 16 bit board providing support for expanded spreadsheets. JRAM AT-3 will be available with two megabytes at $799.

Don't forget the rest of the JRAM family either. We'll continue producing our popular JRAM-2 and our present JRAM AT board, along with our full range of "slim daughter board" modules. The JRAM-2 and JRAM-3 can co-exist in the same computer, but only the JRAM-3 can be used for expanded memory applications. Use JRAM-2 for DOS, multi-tasking, print spooling, and electronic disk.

To complete the family line-up we feature JDISKETTE. This controller card lets you put 1.2 megabyte diskette drive(s) in your PC or XT. You can read, write, and format the AT high density diskettes and communicate easily between a PC or XT and an AT. You won't lose a single slot, since JDISKETTE replaces your old controller and handles up to four internal floppy drives. Free of charge with JDISKETTE, you will receive the popular JFORMAT software, a DOS compatible device driver that gives your computer the ability to handle expanded disk formats.

Welcome to the family!

CIRCLE NO. 197 ON READER SERVICE CARD

TALL TREE SYSTEMS
1120 San Antonio Road
Palo Alto, CA 94303
(415) 964-1980

PC can be made to produce sound with the following commands:

```
SOUND:PROCEDURE OPTIONS(MAIN);
   PUT SKIP LIST(ASCII(7) );
END SOUND;
```

As for program development support, the DRI PL/1 compiler has listing options that can list a program, optionally with interleaved object code, and that can indicate the nesting level of PROC/END and so on. Another option can list the attributes of defined symbols, but this option offers no compiler-produced cross reference listing.

As with full PL/1, DRI PL/1's main weapon for debugging is the PUT statement. Full PL/1 special diagnostic facilities, such as CHECK and SUBSCRIPTRANGE, are omitted, but the basic mechanism of ON ERROR is available. A cryptic reference in one of the manuals says "you may load the symbol table [produced by the linkage editor] under [] SID-86 for debugging;" but, no further explanation of SID-86 is given.

DRI PL/1 error messages are much terser than those produced with mainframe PL/1; they are displayed under the offending line and are usually quite clear. Often it is not necessary to look up messages in the reference manual. In some situations, however, the messages are misleading, requiring some guesswork and experimentation. For example, the error with the following:

```
  215 d              DO I=1 TO 300;
COMP REQ                          ?
  224 c                    END;
COMP REQ                      ?
```

is that the I is a variable within a structure, but this was uncovered only through experimentation.

The compiler does a reasonable job of recovery, continuing to discover further errors rather than stopping, although one error may cause several subsequent errors. All situations that produce messages must be corrected before a linkable .OBJ file is produced. In addition, not all errors are reported. For example,

```
%REPLACE BLEEP BY 'PUT SKIP LIST
   (ASCII(7))';
```

was passed without comment. Only the nonreplacement of BLEEP indicated that something was wrong. Similarly, the ; missing from the first line of

```
DO I = 1 TO 10
   DO J = 1 TO 20;
```

was not reported, but caused errors to be reported from later statements that were actually valid.

FIGURE 1: *Sample RECORD I/O*

```
. . . .
DCL 1 EMPLOYEE,
   2 NAME CHAR(20) VARYING,
   2 DEPARTMENT CHAR (20) VARYING,
   2 HOURLY_WAGE FIXED DECIMAL (5,2),
   2 HOURS_WORKED FIXED DECIMAL (3);
DCL EMP FILE;
DCL KEYVAL BIN FIXED;

OPEN FILE (EMP) UPDATE DIRECT ENVIRONMENT (F(128) )
   TITLE ('A:EMPL');

. . . .
READ FILE(EMP) INTO(EMPLOYEE) KEY(KEYVAL);

. . . .
WRITE FILE(EMP) FROM(EMPLOYEE) KEYFROM(KEYVAL);
```

RECORD I/O is used to read and write data to files without conversion, providing both sequential and direct access to stored files.

Like full PL/1, Subset PL/1 traps most errors and, in the absence of program-defined ON ERROR code, issues a message and aborts the program. Although error messages are generally clear and self-explanatory, the documentation offers no help in tracking down the source code for runtime errors. An error at runtime produces a message such as

```
ERROR(1) "conversion"
Traceback: 07BE 0769 012E 4C00 # 0702
   0322 8090 012E
```

but nothing in the manuals indicated how to relate this to a PL/1 statement.

SYSTEM LIMITS

The code and data sections in the .EXE file created by the linkage editor are each limited to 64KB. Programs may be linked with overlays, to a maximum nesting level of 5, allowing large programs to execute in small memory spaces. The documentation is unclear as to whether the 2*64KB limit applies to each overlay individually or to the maximum program size along an overlay path. A program may address data beyond these limits through the use of BASED data. Doing so makes the total address space available without any performance penalty.

In "Pascal Times Four" (Jeff Duntemann and Michael Bentley, July 1984, p. 58), the same program was compiled with each of the four Pascal compilers and the execution times were measured. This provides a worthwhile comparison of the relative efficiency of the code produced from each compiler.

With just one compiler to evaluate, benchmarks take on a different flavor. Comparative evaluations attempt to say that "all other things being equal . . ." but comparisons among languages are hardly equal. Nevertheless, a general idea of how the object code produced by this compiler compares with that from other languages is important to someone who is considering using PL/1 rather than another language.

Therefore, PL/1 versions of the benchmarks used by Duntemann and Bentley were run, using essentially the same logic and timing methods. These benchmark programs were pentathlon (PENTATH.PLI, listing 4), matrix (MATRIX.PLI, listing 5), and the trigonometry benchmark TRIG.PLI (listing 6). Table 4 lists the results.

The system timer was used for timing. PL/1 has a built-in TIME function that should have been usable, but it returned only blanks. The manual says that "[TIME is] only available if supported by the operating system." But it was easy to call the operating system **get time** function using the DRI-supplied subroutine PCDIO.A86, and it seems a careless omission by DRI not to have provided a subroutine with the same definition as the built-in function.

The DRI PL/1 results compare well with the Pascal and C results (except in the tests with 8087 support); it is safe to say that execution efficiency need not be a crucial concern in selecting PL/1 as a programming language.

In these exercises, PL/1 programs perform "Pascal-like tasks" using algorithms copied closely from Pascal. The

times are therefore relative, because this is not necessarily the way in which the tasks would be performed in PL/1. The I/O benchmark is particularly suspect: efficiency results differed greatly when the program was changed to read and write more than one byte at a time and when it was run without recompilation using a newly-formatted disk.

The results of these tests offer little help in situations where a program is doing more "commercial" tasks involving file processing, so the word extraction program (listing 2) was run. However, this program's time was entirely I/O-related: if the PC had a wait light, it would have been glowing brightly. Because this program does much more processing per record than typical commercial programs, it is safe to assume that PL/1 file-processing programs will run at maximum (I/O limited) speed.

TABLE 4: *PL/1 Execution Speed*

PROGRAM	PC	PC/XT
PENTATHLON		
Benchmark 1[a]	94.2	94.59
Benchmark 2[a]	134.08	134.52
Benchmark 3[a]	24.07	24.33
Benchmark 4[a]	332.46	333.45
Benchmark 5	263.38	210.37
Benchmark 5[b]	92	53
MATRIX[c]	.94	.93
TRIGONOMETRY[d]	298.63	299.98

Execution times in seconds

[a]Time given is for 100 iterations.
[b]Same as benchmark 5, but data are read/written 250 characters at a time.
[c]Time to execute SQUARE procedure.
[d]Time to calculate 10,000 tangents.

The DRI PL/1 compares well with compilers for other languages (specifically, Pascal and C), except in the tests with 8087 support; it is safe to say that execution efficiency need not be a crucial concern in the selection of PL/1.

COMPILING FULL PL/1?

To better understand the differences between mainframe PL/1 and DRI PL/1, a large mainframe program was copied to the PC. The program was ADVENTURE, a computer game that has been passed around informally among programmers for several years, in which the player attempts to retrieve treasure from a cave. Remember that DRI makes no claim that its compiler is compatible with full PL/1; indeed the documentation says, "Subset G has the formal structure of the full language, but in some ways it is a new language . . ."

The reasons this program was selected were several:

- ADVENTURE is a large program (3,200 lines), and because the code was unfamiliar, it provided a good off-the-shelf test of the runtime diagnostics.
- The code does not exploit the more advanced features of PL/1. In fact, the program may have been written originally in some other language, probably FORTRAN; therefore few changes should have been necessary.
- The program used no redefined or BASED data and was therefore likely to be free of hidden implementation dependencies, such as assumptions that pointers take four bytes.
- It was readily available, being a great favorite in the household.

The first step was to write a short PL/1 program to process the source code, removing excess trailing links and the leading control character and trailing sequence numbers from the card-image formatted mainframe file. This halved the size of the text to about 130,000 bytes and allowed it to be edited. Then the text was edited to make it valid by Subset G rules and to replace calls to mainframe assembly language routines with standard PL/1 code. After a few attempts, the program compiled cleanly. However, when the program was run, it was apparent that the logic depended on one of the changes that had been made, and it was necessary to understand the program logic in some detail in order to program it. Although this task remains incomplete, it advanced far enough for me to draw some conclusions.

Debugging large programs successfully requires a cross-reference report from the compiler (which the DRI compiler does not provide). Additionally, a way of locating the statement from which an error arises is needed. Although a lack of full PL/1 options such as GOSTATEMENT and FLOW was anticipated, it was expected that the failing statement could be located from the traceback printed as part of the standard error action. Instead, this step required guesswork and the insertion of a diagnostic PUT statement to confirm the guess. With compile times for this program approaching ten minutes, extra compilation was unwelcome

This exercise has shown that a conversion from full PL/1 in the mainframe environment to Subset PL/1 on a PC should be treated as a program development with phased testing and implementation, rather than as a mechanical conversion in which everything is simply recompiled and then works automatically. Although a smooth conversion might be expected when a well-structured program allows a phased top-down approach, with a monolithic unstructured program the relative paucity of DRI runtime diagnostic help necessitates a compile-and-pray loop. At this point it is apparent that ADVENTURE, and programs of its relative size and complexity, would have to be broken down into smaller units and each unit tested individually.

A MIXED BAG

Users demand a great deal from manuals and with good reason. Sound documentation is vital if new users are to become productive quickly: moreover, lasting impressions of product quality can depend on the documentation. Two distinct needs must be addressed: learning and reference. DRI meets these, respectively, in its two manuals, the PL/1 user's guide and the reference manual. Together they comprise approximately 500 pages in a standard DRI gray-brown slipcase binder. Although they are not typeset, variations in type size, in combination with diagrams and the use of color, produce a nice effect.

The manuals are generally good for learning. The user's guide is particularly strong, working from simple concepts and programs to commercial processing, dynamic memory techniques, and overlays. It also goes through a series of examples, which also are provided on one of the disks. On the down side, chapter 6 (which explains how to install and run the compiler) would be better placed at the beginning of the manual. The manual would be improved further with the addition of chapters on program debugging and on

interfacing with PC-DOS. Much of the material in the PC-DOS chapter is also on the sample disks. Chapters that provide programmers who are experienced in other languages with quick insights into PL/1 also would be useful.

As references, however, the manuals range from barely adequate to abysmal. The deficiencies, particularly of the reference manual, quickly become apparent when a specific question needs answering. For example, trying to find the precise rules and alternatives for a particular statement (a standard search during diagnostics) is surprisingly difficult because no single chapter offers a full definition of each statement.

Answering particular questions requires the use of both manuals; sometimes scanning the indexes will unearth a likely-looking synonym. A user may often have to find an answer through experimentation. Finally, the removal of reference to the product's S-bit derivation was sloppy; for example, the table of ASCII codes only goes to 127.

However, in the final analysis, this compiler pleases more than it disappoints. The DRI PL/1 compiler's worst faults are in its documentation and its diagnostic facilities, both of which (certainly the documentation) can be fixed without a major rewrite of the software.

Users choosing a language can seriously consider using PL/1 with this compiler if the features of the language support such a decision. The user need not worry about hidden rules and restrictions in a particular implementation that negate its theoretical advantages. Users looking to transfer their mainframe PL/1 experience to PC programming will find the essential features of PL/1 available in the DRI implementation. Although adjustments are necessary and it is not possible to simply download a mainframe PL/1 program and recompile it to have a PC version, the major adjustments for programmers are probably going to be in environmental factors, such as I/O concepts, rather than in language dialect differences.

With its presentation and price ($750), the DRI compiler is aimed at the professional market. Although it is

Users looking to transfer their mainframe PL/1 experience to the PC will find that DRI PL/1 has the essential features of the language.

more expensive than other language compilers, if the feature list matches a programmer's requirements, this PL/1 compiler could be a good value. PL/1 can do anything that C, Pascal, Modula, BASIC, COBOL, or FORTRAN can do, and as elegantly (in general), without nonstandard extensions. With the DRI compiler filling in the lower end, PL/1 is available now for computers ranging from micros to the largest mainframes. Do not ignore it.

PL/1 compiler 1.0: $750
Digital Research, Inc.
P. O. Box DRI
Monterey, CA 93942
408/649-3896

CIRCLE 341 ON READER SERVICE CARD

Robert Barnes is managing director of MANA Systems Limited in Northcote Auckland, New Zealand, a company that produces 4GL tools for IBM and FACOM systems. He is the author of PL/I for Programmers *(Elsevier, 1979).*

New Version

Sizzling C.

The fastest C. The C that Microsoft developed to write its own software programs. Hot.

So hot that we can make this claim: Virtually every program runs faster with Microsoft® C Compiler than with any other MS-DOS® C compiler.

Efficient C.

We give you everything you need to write code so tight your computer will scream.

"Preliminary testing on the Microsoft C Compiler produced code that was significantly smaller than that produced by other C compilers."
Paul Springer, Ashton-Tate.

"We found the FAR pointer very helpful for situations where a mix of memory models offers the greatest efficiency."
Robert Frankston, Software Arts.

"The portability of the code between MS-DOS and XENIX® is great."
Jim Bean, Peachtree Software.

But it's really no surprise that our C stretches your micro to its limits. We wrote both the MS-DOS and the XENIX operating systems.

For the name of your nearest Microsoft dealer, or to upgrade from Microsoft C Compiler or Lattice C, call (800) 426-9400. In Washington State, Alaska, Hawaii and Canada, call (206) 828-8088.

And make your programs really cook.

MICROSOFT®
The High Performance Software™

Microsoft C Compiler Version 3.0

Microsoft C Compiler
- Produces compact code and fast executables.
- Implements register variables.
- Small, Medium and Large Memory model Libraries-Mix models with NEAR and FAR pointers.
- Transport source and object code between MS-DOS & XENIX operating systems.
- Library routines implement most of UNIX System V C library.
- Choose from three Math libraries and generate in-line 8087/287 instructions or floating point calls.
  - –Floating point emulator (utilizes the 8087/287 if installed).
  - –8087/287 coprocessor support.
  - –Alternate math package–extra speed on systems without an 8087/287.
- Link routines written in Microsoft FORTRAN (V 3.3 or higher), Microsoft Pascal (V 3.3 or higher) or Microsoft Macro Assembler.
- Supports MS-DOS pathnames and Input/Output redirection.
- File sharing and record and file locking is supported.
- Do source level debugging, with the Symbolic Debug Utility, available separately with the Microsoft Macro Assembler Package.*

Library Manager
Create, organize and maintain your object module libraries created with Microsoft languages.

Object Code Linker
- Simple overlay linker combines relocatable object modules created using Microsoft Languages into a single program.
- Link very large programs (over IMB, using overlays).

EXEPACK Utility
A new utility to compress sequences of identical characters from an executable file and to optimize the relocation table.

EXEMOD Utility
A new utility used to modify the fields in the header according to the instructions given by the user in the command line.

'C' Benchmarks–done on a Compaq Plus with 512k memory with no 8087. Program "SIEVE," with register variables,

	Exec Time	Code Size	EXE Size
Microsoft C	:9.39	141	5,914
Lattice C	:12.24	164	20,072

*Purchase both Microsoft C Compiler and Microsoft Macro Assembler and get a $25 rebate direct from Microsoft. See package for details.

LISTING 1: HELLO.PLI

```
/*  Small program to print hello  */
/*  Robert Barnes  */


demo:procedure options(main);

    put list ('hello');

end demo;
```

LISTING 2: KWIC.PLI

```
kwic:proc options(main);
  /* KWIC (Keyword in Context) system.  */
  /* create or augment list of noise words, ie words which
      are ignored for indexing, setting them up in an ordered
      tree structure, eliminating duplicates */
dcl (infile,outfile) char(14) var;
dcl (incount, outcount,i,j,k) bin fixed;
dcl  in file,
     out file;
dcl  inrec char(20) varying;
dcl  (inline,outrec) char (250) varying;
%include 'wordstr';

PUT list ('KWIC (Keyword-in-context) system');
Put skip list ('build or augment list of noise words.');

PUT skip list ('First read list of predefined noise words.');
PUT skip list ('Enter name of input file ');
GET edit(infile)(A);
PUT skip list ('Enter name of output file ');
GET edit(outfile)(A);
put skip(2);
open file(in) stream input title(infile);
open file(out) stream output title(outfile) print;
on endfile (in) begin;
 put skip(3) list('number of records input = ',incount);
 go to exit;
end;

/* general initialization */
start=null;
p=addr(start);
ioloop:
do incount = 0 by 1;
 get file(in) list (inrec);
 put list (inrec);
 call find;
end ioloop;
exit:
close file(in);

put skip list ('enter file to be scanned for more words ');
get list (infile);
open file(in) title(infile) stream input;
on endfile (in) begin;
 put skip(3) list('number of records processed = ',incount);
 go to exit2;
end;

ioloop2:
do incount = 0 by 1;
 read file(in) into (inline);
 /* remove all special characters and convert to upper case */
 outrec=translate(inline,

                                  ABCDEFGHIJKLMNOPQRSTUVWXYZ
ABCDEFGHIJKLMNOPQRSTUVWXYZ      ');
```

New Version

Mighty Macro Assembler.

The new Microsoft® Macro Assembler package. A complete development environment that makes you a more productive programmer. Whether you're using Macro Assembler or any Microsoft high level language.

A common calling convention lets you easily call assembly language routines from any high level Microsoft language to add an extra burst of blinding speed.

Better Debugging.

The new Symbolic Debug Utility lets you stay close to the source. Now you can step through your assembled or compiled code by name rather than by address. Source level display for Microsoft Pascal, FORTRAN, and C allows you to view both your original source and the resulting code.

And we stuffed our package with a full set of the most useful utilities around. So that you can link, maintain and organize your programs like never before.

Who else but Microsoft could build so much into one package for $150?

For the name of your nearest Microsoft dealer call (800) 426-9400. In Washington State, Alaska, Hawaii and Canada, call (206) 828-8088. And if you're already using Microsoft or IBM® Macro Assembler, ask us how you can upgrade to the mightiest Macro of them all.

MICROSOFT®
The High Performance Software™

Microsoft Macro Assembler Package:

Macro Assembler
- For the 8086/8087/8088 and now the 186/286/287.
- Define macros.
- Conditional Assembly.
- Case sensitivity for symbols.

New Interactive Symbolic Debug Utility
- Controlled testing environment for debugging.
- Source line display of Microsoft FORTRAN, Pascal and C Programs.
- Set breakpoints on line numbers and symbols.
- Single step to follow program execution.
- Disassemble object code.
- Display values.
- Make minor changes without reassembling.

New Program Maintenance Utility
- Rebuilds your applications after your source files have been changed.
- Similar to UNIX™ Make utility.

Library Manager
- Create, organize and maintain your object module libraries created with Microsoft Languages.
- Set page size (default of 16 bytes).

Object Code Linker
- Simple overlaying linker combines relocatable object modules created using Microsoft Languages into a single program.
- Load Map generation.
- Specify from 1 to 1024 segments.

Cross Reference Utility for the Macro Assembler
- Creates a cross-reference listing of the definitions and locations of all symbols used in an assembly language program.

```
PUT skip list(inline);
/* pick out each word */
j=1;
do while(j<length(inline));
 do j = j to length(inline) while(substr(outrec,j,1)=' ');
 end;
 do k = j to length(inline) while(substr(outrec,k,1)^=' ');
 end;
 if j<length(inline)
 then do;
       inrec=substr(outrec,j,k-j);
       call find;
      end;
 j=k+1;
end;
end ioloop2;
signal endfile(in);
exit2:
/* logical end-of-program */

call traverse(start);
/* logical end of program */


find:proc;
 /* find word in binary tree */
 p = start;
 p2=addr(start);
 do while(p^=null&word^=inrec);
  if word<inrec
  then do;
        p2=addr(higher);
        p=higher;
       end;
  else do;
        p2=addr(lower);
        p=lower;
       end;
 end;
 if p=null
 then do;
       allocate wordlist set(p);
       link=p;
       lower=null;
       higher=null;
       word=inrec;
      end;
 end find;
 traverse:proc(pstart) recursive;
  dcl pstart ptr;
  %include 'wordstr';
  p = pstart;
  if lower^=null
  then call traverse(lower);
  put list(word);
  put file(out) list(word);
  if higher^=null
  then call traverse(higher);
 end traverse;
end kwic;
```

LISTING 3: WORDSTR.PLI

```
dcl (start,p,p2) ptr,
    1 wordlist based (p),
      3 (lower,
         higher) ptr,
      3 word char(20) var;
dcl link ptr based(p2);
```

Does your C compiler understand you're only human?

Mark Williams knows that programmers are like everyone else: you tend to put your pants on one leg at a time.

But you still have to get your programs up and running as fast as possible. With all the buttons buttoned and all the zippers zipped.

That's why we developed the C Programming System. So you don't get caught with your pants down.

To err is human–to debug, superhuman.

Normally, nothing is more frustrating for a programmer than the debugging process. You've spent months just getting the code written, but you know it's going to take at least that much more time to get the program running right.

That's where our C Source Debugger (*csd*) can be a big help.

csd lets you debug like a human being–in C, not assembler–looking right at your code through the *csd* window, an exclusive Mark Williams feature. You can set trace-points to stop program execution at particular program lines, trace and display the value of any C expression or variable, and much more.

With *csd* you can run the target program a line at a time, continue to the next tracepoint, or even restart the whole program right in the middle of debugging. Meanwhile, you're squashing bugs as you find them. And your program will run without modification.

Get a leg up on the competition.

Every company says its compiler produces the fastest, densest code. But Mark Williams actually proves it. Take a look at the benchmark tests below and see if you don't agree.

Now imagine just how much more competitive this kind of performance could make your products.

The C Programming System supports the complete C language as defined by Kernighan & Ritchie. But it also goes on to include void and enumerated data types, register variables, structure assignments, Berkeley structure rules, and the biggest C library available. With support for a wide variety of third-party C libraries and utilities.

You also get MS-DOS compatibility, large and small memory models, 8087 in-line support, and one-step compiling. A full range of options increases your flexibility, letting you compile without linking, link without compiling, and more.

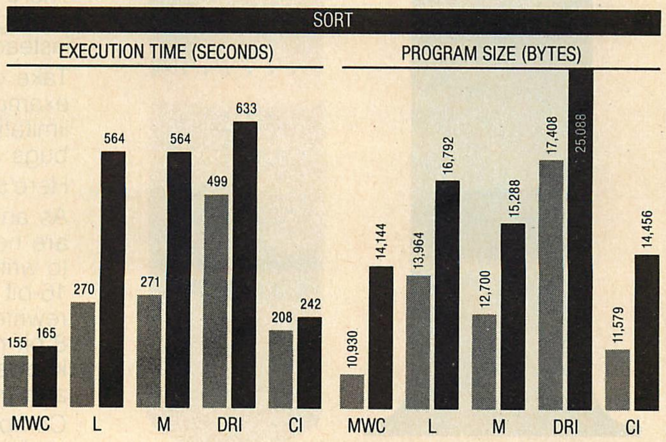
With all these advantages, it's no wonder Intel, DEC, Wang, and many others have made MWC86 their compiler of choice. (After all, they're only human.)

A human interest story with a happy ending.

All right, you're interested–which proves you're not only human, but smart. So what do you do now?

Easy. Just call 1-800-MWC-1700. You'll talk to another human being who'll answer any questions you have. And if you want to order, we'll send you the complete system, including MWC86 compiler, *csd* debugger, complete library of functions, and more. All for just $495.

The sooner you call, the sooner you can be winning the race to get your products out the door. Which is, after all, a very human race.



■–Small Memory Model
■–Large Memory Model

NOTE: Sort program as in Byte, August 1983, p. 91. Register declaration added. Further information on these benchmarks available from Mark Williams Company upon request.

Mark Williams Company
1430 West Wrightwood Avenue
Chicago, Illinois 60614

Call 1-800-MWC-1700 To order.

IN ILLINOIS CALL 312-472-6659. VISA/MC ACCEPTED

CIRCLE NO. 186 ON READER SERVICE CARD

LISTING 4: PENTATH.PLI

```
/*  The Pentathlon Program
    Translated to PL/1 from the PASCAL version by Mark
    Townsend & Robert Barnes.  The PASCAL versions were
    translated from the C version of these routines by
    Michael Brian Bentley. Originally written by William
    J. Hunt for PC-TECH Journal readers everywhere.
 */

pentathalon: proc options (main);

dcl (i,niter,ibench) bin fixed(15);
 niter=100;
 /* main program */
 do ibench=1 to 4;
   put skip edit('100 iterations ', ibench)(skip,a, f(3));
   put list('starting ');
   call puttime;
   do i = 1 to niter;
     if ibench=1 then call bench1;
     if ibench=2 then call bench2;
     if ibench=3 then call bench3;
     if ibench=4 then call bench4;
   end;
   put skip list('finished ');
   call puttime;
 end;
 put skip list ('create file for benchmark 5');
 call makefile;
 put skip list('starting I/O benchmark');
 call puttime;
 call bench5;
 put skip list('finished ');
 call puttime;


bench1:procedure; /* floating point arithmetic benchmark */
dcl (i,j) bin fixed;
dcl (x,y) (0:99) float,
     z float;

do i = 0 to 99;
  x(i) = i + 1;
  y(i) = 3 + i;
end;
z = 0;
do j = 0 to 9;
  do i = 0 to 99;
    z = z + x(i) * y(i);
  end;
end;
end bench1;
bench2:procedure; /* function calling benchmark */

dcl i bin fixed(15);

do i = 0 to 19999 ;
   call dummy((i));     /* calls a dummy procedure */
                        /* i doesn't change        */
end;

dummy:procedure(pi);
dcl pi bin fixed(15);
pi = pi + 1;
end;
end bench2;

bench3:procedure;      /* string copy benchmark */

dcl  i bin fixed;
dcl  s(500) char(1);
dcl  s2(500) char(1);

do i=1 to 499;
  s(i)='a';
end;
```

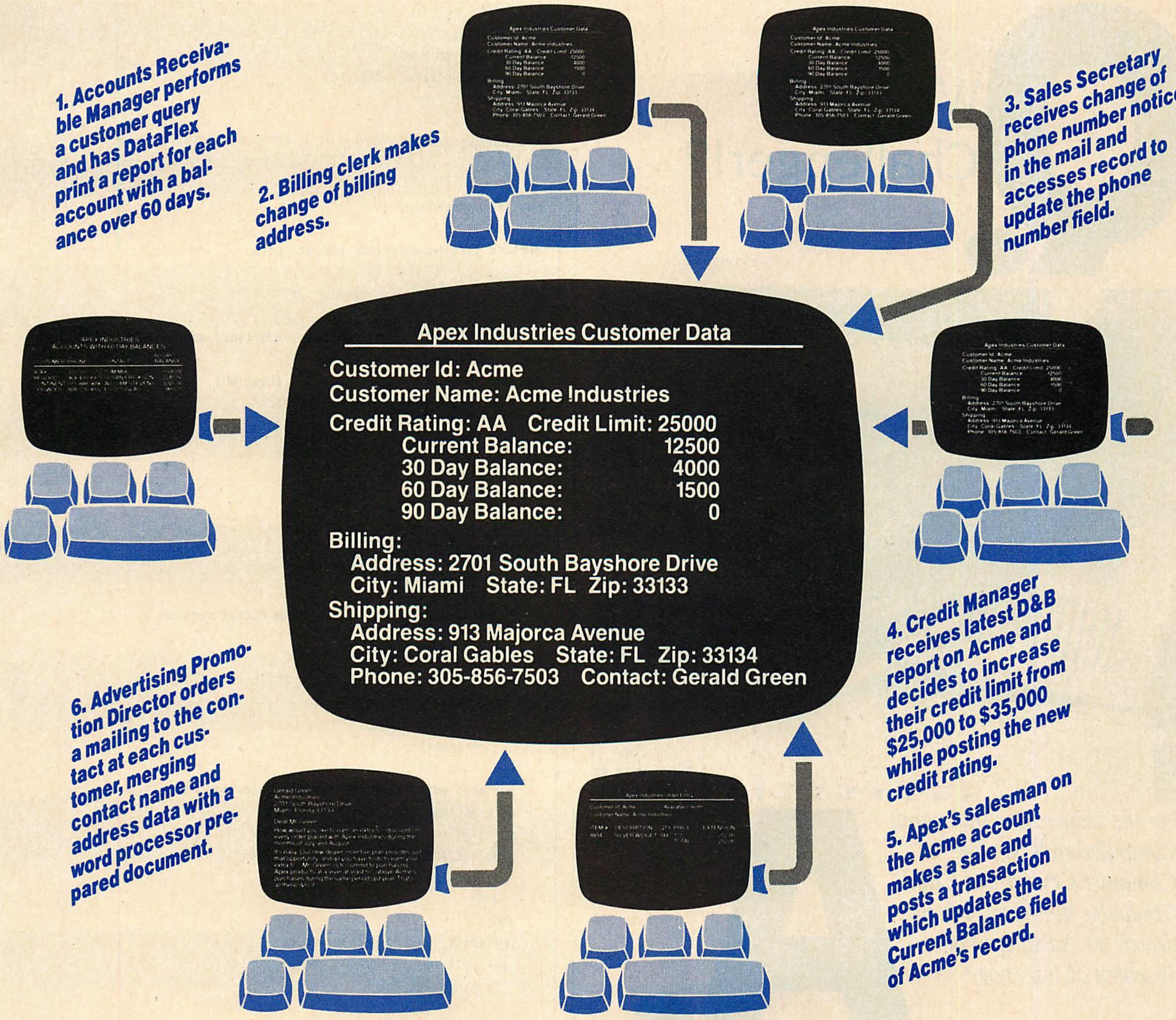
```
s(500)=ascii(0);

do i = 1 to 100;
  s2=s;
end;
end bench3;

bench4: procedure ;        /* character count benchmark */
dcl  i  bin fixed(15);
dcl  s(500) char(1);
dcl  cnt(0:255) bin fixed;

/* bench4 - initialize string array for counting */
do i = 1 to 500;
  s(i) = ascii(i);
end;

do i = 1 to 100;
  call count_char(s,cnt);
end;

count_char:proc(strng,counts);
dcl strng(500) char(1);
dcl counts(0:255) bin fixed;
dcl  i bin fixed(15);
dcl  c char(1);
dcl  idx bin fixed;

do i=1 to 500;
  idx = rank(strng(i));
  counts( idx ) = counts( idx ) + 1;
end;
end count_char;

end bench4;

bench5:proc;        /* file copy with getc/putc */

dcl n bin fixed(15);
dcl (infile,outfile) file;
dcl data char(1) var;
open file(infile) stream input title('test.in' );
open file(outfile) stream output print title('test.out' );
n = 0;
on endfile(infile) goto exit;
do while('1'b);
  n = n + 1;
  get file(infile) edit(data)(a(1));
  put file(outfile) edit(data)(a(1));
end;
exit:put skip;
put edit(n,' characters')(skip,f(7),a);
close file(infile);
close file(outfile);
end bench5;

makefile:proc;           /* create a test file */
        dcl  victim file;
        dcl  n bin fixed(15);

        /* makefile */
        open file(victim) stream output print title('test.in');
        do n = 0 to 29999;
         put file(victim) edit('a')(a(1));
        end;
        close file(victim);
end;
puttime: procedure;
declare gettime entry (pointer,pointer,pointer,pointer);
declare (hour,min,sec,fraction) fixed(7);

call gettime(addr(hour),addr(min),addr(sec),addr(fraction));
put skip list('The time is now ');
put edit(hour,':',min,':',sec,'.',fraction)
         (f(2),a,f(2),a,f(2),a,f(2));
end puttime;

end pentathalon;
```

LISTING 5: MATRIX.PLI

```
/* procedure matrix                                              */
/* translated by Mark Townsend                                   */
/* from: PASCAL PROGRAMS FOR SCIENTISTS AND ENGINEERS            */
/* by Alan R. Miller (c) 1981 by Sybex, Inc.                     */

matrix:proc options(main);

   dcl y(20) bin float;
   dcl g(20) bin float;
   dcl x(20,20) bin float;
   dcl a(20,20) bin float;
   dcl (nrow,ncol) bin fixed;
   dcl ch char(1);

/* start of procedure matrix                                     */
   call get_data(x,y,nrow,ncol);
   put skip edit('>>press any key then return to start: ')(a);
   get edit(ch)(a(1));
   put skip edit('>>starting matrix inversion..')(skip,a);
   call puttime;
   call square(x,y,a,g,nrow,ncol);
   put edit('..all done!')(skip,a);
   call puttime;
   call write_data;

get_data:proc(x,y,nrow,ncol);
   dcl x(20,20) bin float;
   dcl y(20) bin float;
   dcl (nrow,ncol) bin fixed;
   dcl (i,j) bin fixed;

/* get values for NROW,NCOL and arrays x and y                   */

   nrow=10;
   ncol=10;
   do i=1 to nrow;
      x(i,1)=1;
      do j=2 to ncol;
        x(i,j)=i*x(i,j-1);
      end;
      y(i)=2*i;
    end;
end get_data;

write_data:proc;
   dcl (i,j) bin fixed;

/* write out the answers                                         */
   put skip edit('       x')(a);
   put edit (':  y  ')(col(71),a);
   do i=1 to nrow;
    put skip;
    do j=1 to ncol;
      put edit(x(i,j),' ')(E(6),a(1));
     end;
     put edit(':',y(i))(col(71), a, e(6));
   end;
   put skip edit('       a')(skip,a);
   put edit (':  b')(col(71),a);
   do i=1 to nrow;
     do j=1 to ncol;
       put edit(a(i,j),' ')(e(6),a(1));
     end;
     put edit(':',g(i))(col(71),a(1),e(6));
   end;
 end write_data;

 square:proc(x,y,a,g,nrow,ncol);
   dcl (x,a)(20,20) bin float;
   dcl (y,g)(20) bin float;
   dcl (nrow,ncol) bin fixed;
   dcl (i,k,l) bin fixed;

/* matrix multiplication routine                                 */
/* a = transpose x times x                                       */
```

ALL AT ONCE!

AND NEVER A "LOCKED OUT" USER!



DataFlex is the only application development database which ***automatically*** gives you true multi-user capabilities. Other systems can lock you out of records or entire files for the full time they are being used by someone else. DataFlex, however, locks only the data being changed, and ***only*** during the micro-seconds it takes to actually write it to the file! The updated record is then immediately available. The number of users who can access, and change, records at the same time is limited only by the number of terminals on your system or network. Call or write today for all the details on DataFlex... the true multi-user database.

DATAFLEX™

DATA ACCESS CORPORATION.
8525 SW 129 Terrace, Miami, FL 33156 (305) 238-0012
Telex 469021 DATA ACCESS CI

Compatible with IBM PCDOS 1.x, 2.x, 3.x, MSDOS 1.x, 2.x, 3.x, IBM "AT" XENIX, IBM PC NETWORK, CP/M, CP/M-86, Concurrent CP/M-86, Concurrent DOS, MP/M-86, Novell NetWare, TurboDos, Corvus w/IBM PC, Molecular N/Star, Televideo InfoShare, 3Com EtherShare, PC-Net, NorthStar Dimension, Action DPC/OS, DMS Hi-Net, Alloy Engineering RTNX.

CIRCLE NO. 118 ON READER SERVICE CARD

```
/* g = y times x                                    */

   do k=1 to ncol;
     do l=1 to k;
       a(k,l)=0;
       do i=1 to nrow;
         a(k,l)=a(k,l)+x(i,l)*x(i,k);
         if k^=l
         then a(l,k)=a(k,l);
       end;
     end;
     g(k)=0;
     do i=1 to nrow;
       g(k)=g(k)+y(i)*x(i,k);
     end;
   end;
 end square;


 puttime: procedure;

 declare gettime entry(ptr,ptr,ptr,ptr);

 declare (hour,min,sec,fraction) fixed(7);

 call gettime(addr(hour),addr(min),addr(sec),addr(fraction));
 put skip list('The time is now ');
 put edit(hour,':',min,':',sec,'.',fraction)
        (f(2),a,f(2),a,f(2),a,f(2));

 end puttime;

 end matrix;
```

LISTING 6: TRIG.PLI

```
/* procedure trig                                    */
/* translated by Robert Barnes from Pascal version   */

trig:proc options(main);

   dcl tans(100) float;
   dcl (r,t) bin fixed,
       ch char(1);

   put skip edit('starting tangent calculations ..')(skip,a);
   call puttime;
   do t = 1 to 100;
     do r = 1 to 100;
       tans(r)=sin(r)/cos(r);
     end;
   end;
   put edit('..all done!')(skip,a);
   call puttime;


puttime: procedure;

declare gettime entry(ptr,ptr,ptr,ptr);

declare (hour,min,sec,fraction) fixed(7);

call gettime(addr(hour),addr(min),addr(sec),addr(fraction));
put skip list('The time is now ');
put edit(hour,':',min,':',sec,'.',fraction)
       (f(2),a,f(2),a,f(2),a,f(2));

end puttime;

end trig;
```

INTRODUCING THE MOST RADICAL ALGORITHM IN DATABASE HISTORY.

COMMON SENSE.

COMMON SENSE IN DATABASE MANAGEMENT
NUMBER ONE IN A SERIES.

IT'S NOT HOW HARD YOU WORK. IT'S HOW MUCH YOU GET DONE.

There's an odd contradiction built into most database software. Most of the features that make it powerful enough to do the job make it a real struggle to use.

But at Microrim,® we've never thought it makes sense to do things the hard way. That's why we're introducing R:base™ 5000. With a brand new feature that lets you create programs up to ten times faster.

CLIENT DATABASE MENU
1. Add a client master list.
2. Change a client master record.
3. Delete a client master record.
4. Print client report.
5. Exit.

PRODUCTIVITY:
R:base 5000 vs. dBASE III

Using R:base 5000 and dBASE III,™ we built this menu and linked it to its subroutines. The resulting applications were equivalent. The effort required wasn't.

	R:base 5000	dBASE III
Keystrokes	434	6588
Command Lines	47	244
Automated steps	37	0
Time*	9 minutes	2 hours

**Your actual time may vary depending on skill level*

THE APPLICATION EXPRESS.™ WHAT A DIFFERENCE.

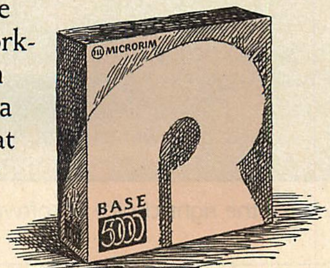
Since the Application Express automatically generates programming code, it can give you the tremendous advantage we've documented in the attached chart. If you've never programmed before, these automated steps can make all the difference between getting the job done and giving up completely. And if you're an experienced programmer, the Application Express can give you ten to one productivity gains. Of course, R:base 5000 also gives you the powerful procedural language and report writer it takes to create highly customized applications.

SEE FOR YOURSELF: 1-800-547-4000.

If you believe in common sense as much as we do, you won't take our word for it; you'll get your hands on a copy and make up your own mind. And that's just what we'd like you to do. For only $9.95 (plus shipping) we'll send you a mini-version of the product that lets you build real-life applications.

Just call 1-800-547-4000 and ask for Dept. 990 . From Oregon, or outside the U.S., call 1-503-684-3000, Dept. 990 . We'll send your copy right out. If you'd like to see R:base 5000 today, head straight for a leading software store or computer dealer. If you own R:base 4000, ask your dealer for a trade-up kit.

We'll show you how easy it is to get a handle on your workload. When you've got a progam that does things your way.



R:BASE 5000
FROM MICRORIM

IT ALL COMES DOWN TO COMMON SENSE.

dBASE III is a trademark of Ashton-Tate.

CIRCLE NO. 193 ON READER SERVICE CARD

You know that choosing the right software is serious business. So does WATCOM.

So before you make any decisions about your software needs, talk to WATCOM— the people major software users around the world have trusted for years.

WATCOM has the products you need to get the job done right. Proven performers like WATFOR*, WATFIV*, WATBOL*, and SCRIPT. Plus new leaders in software for PC workstations and micro-to-mainframe communications. Networks, language interpreters and compilers. Text preparation and data management. All WATCOM products are human engineered to provide the optimum in people efficiency and productivity. And they're designed to run compatibly on IBM mainframes and PC's, Digital mainframes and micros, and Commodore micros.

Whatever you need is backed up by WATCOM's innovative maintenance and support services. You'll be kept up to date with the latest in product enhancements and information. And our publications and seminars will help you get the most out of your software investment.

WATCOM. Quality products. Professional service. And a reputation built on more than 150,000 licensed mainframe and micro software programs throughout the world. So talk to us before you decide. After all, choosing the right software is serious business. For you. And for WATCOM.

Make the Right Choice: WATFOR-77 — PC and 370

Optimizing programmer productivity makes WATFOR-77 the right choice.

WATFOR-77 significantly reduces the time required for program development by compiling and executing the program directly in the computer memory in one step. Together with excellent debugging and error diagnostics, WATFOR-77 optimizes programmer efficiency and overall productivity.

WATFOR-77 is the newest member of the WATFOR family of debugging FORTRAN compilers and is ideal in business, educational, and scientific applications. Supporting the full FORTRAN 77 language, WATFOR-77 is now available for the IBM PC* DOS and 370 VM/SP CMS*.

Make the right choice. Call or write WATCOM today and we'll tell you all about WATFOR-77 or any of WATCOM's other people-efficient products.

WATCOM

The right choice in software.

Yes! I want to make the right choice in software. Send me more information on: ☐ WATFOR 77 ☐ WATCOM Software Catalogue

Name:____________________

Company:____________________

Title:____________________

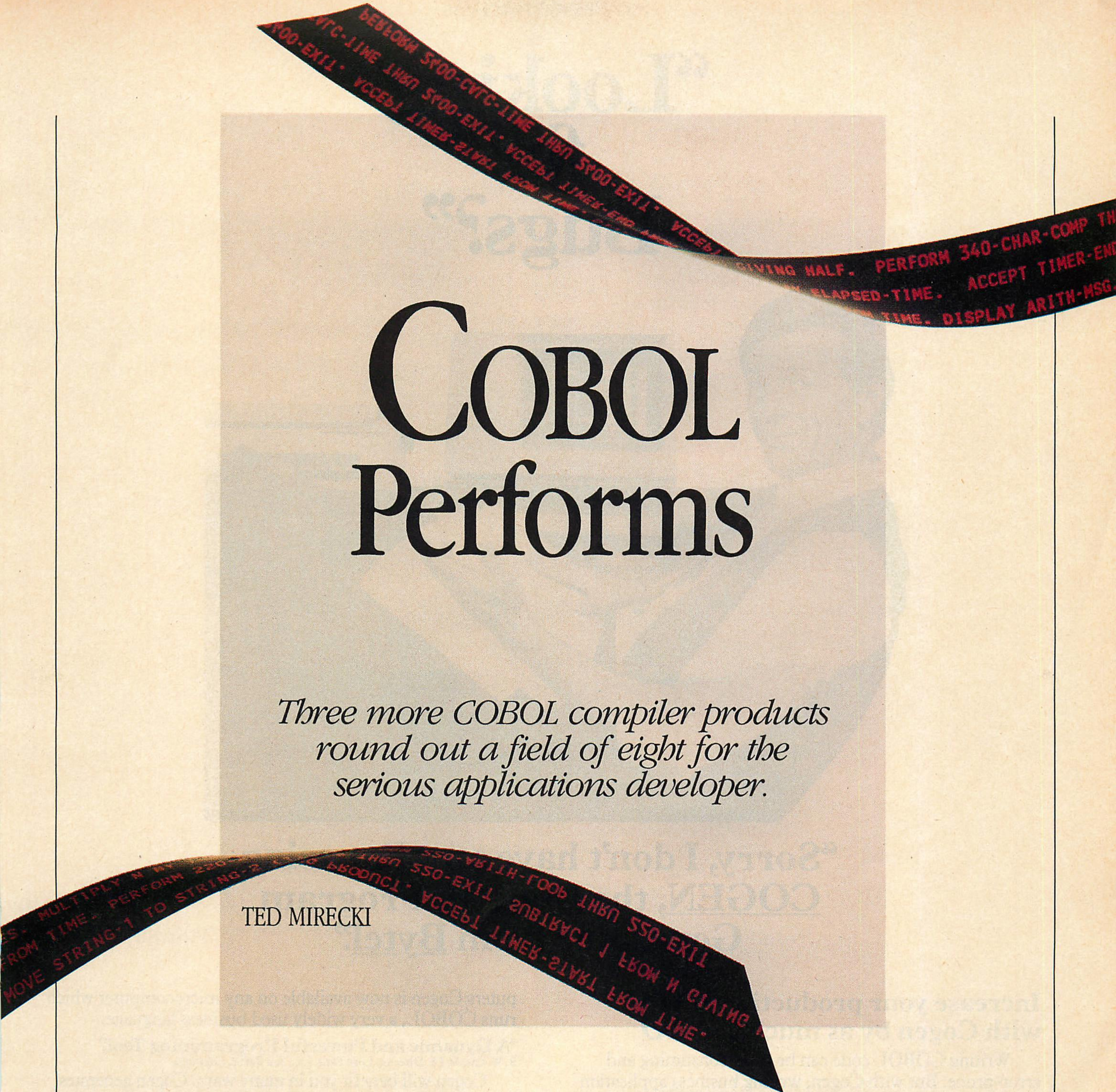
Address:____________________

City:__________ **State:**__________ **Zip:**__________

WATCOM PRODUCTS INC.
415 Phillip Street
Waterloo, Ontario, Canada
N2L 3X2
(519) 886-3700
Telex 06-955458

* WATFOR, WATFIV and WATBOL are registered trademarks of the University of Waterloo.
* IBM PC and IBM 370 VM/SP CMS are registered trademarks of International Business Machines Corporation.

CIRCLE NO. 213 ON READER SERVICE CARD



COBOL Performs

Three more COBOL compiler products round out a field of eight for the serious applications developer.

TED MIRECKI

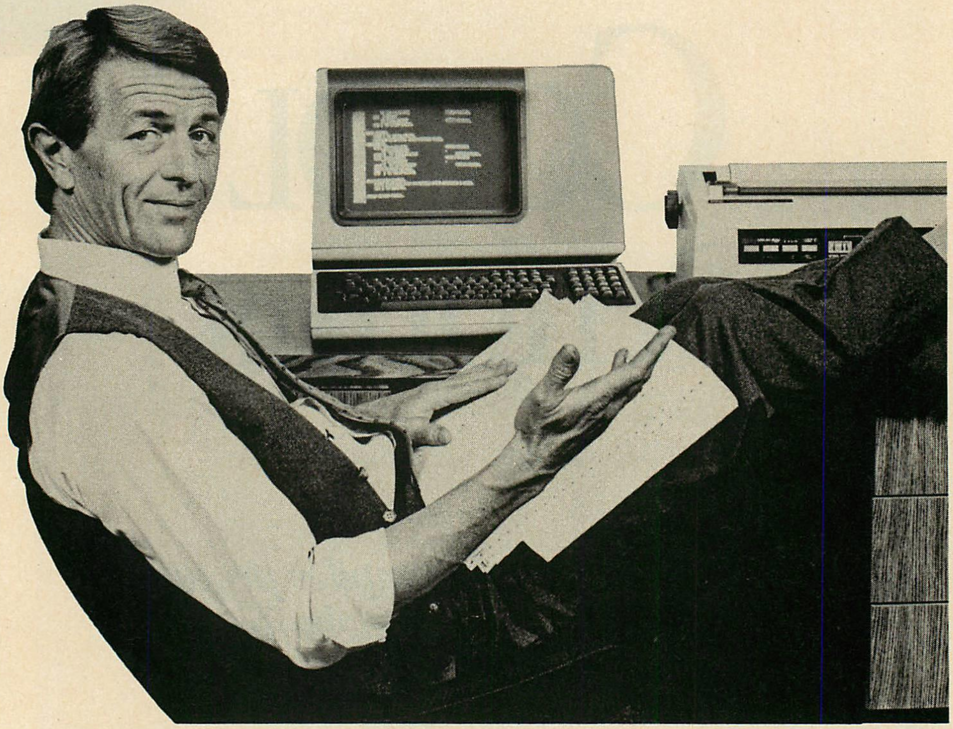
This last article in the series on COBOL compilers presents three more product reviews, plus summary evaluations of all eight of the compilers discussed in the series. To facilitate comparisons among the eight, the tables in this article include results for the previously reviewed compilers in addition to the data for the three products reviewed here: IBM COBOL 1.0, mbp COBOL 9.0, and Micro Focus Professional COBOL.

Micro Focus Professional COBOL is a repackaging of the company's Level II compiler with a series of utilities into an integrated development environment. The compiler itself was reviewed in the July issue (see "COBOL Performs," Ted Mirecki, July 1985, p. 111). The discussion here concentrates on the characteristics of the surrounding environment, so this product is reviewed separately, following the IBM and mbp compilers.

IBM COBOL 1.0 is almost as old as the IBM PC and just as outdated. Certified at the low-intermediate level, it was not reviewed with its low-intermediate peers because of the reputed imminence of a new version. At press time, however, 1.0 was the only COBOL available from this major player.

PHOTOGRAPH • MARC DAVID COHEN

“Looking for Bugs?”



“Sorry, I don’t have any. I’m using COGEN, the COBOL Program Generator from Bytel.”

Increase your productivity with Cogen by as much as 10

Writing COBOL code can be time-consuming and error-prone. But with Cogen, writing business application programs is a breeze. Cogen is a automated Program Generator which produces bug-free COBOL code. You interact with Cogen through menus, prompts and data-entry screens. Once you’ve defined your programming task to Cogen, it does the rest...no more repetitive keying and other drudgery. You can create independent modules to link with your own programs, or you can have Cogen create complete application programs for file maintenance, inquiries and reports. You will be amazed at the combination of power and ease of use embodied in Cogen. Having proved itself on minicomputers Cogen is now available on any microcomputer which runs COBOL, a very widely used business language.

“A Dynamic and Powerful Programming Tool.”
R. W. Lay, M.I.S. Director, Logo Paris, Inc., San Rafael, California.

Cogen will benefit you in many ways. Cogen generates formal, structured programs automatically, so your code is standardized, self-documenting, efficient and easy to maintain. “Screen painting” techniques let you draw the screens exactly as you want to see them displayed by your program, accurately, efficiently and quickly. Cogen has extensive report writing facilities. Its menu driven format produces dozens of lines of bug-free code with just a few keystrokes, so your new programmers can get started much sooner. And using Cogen means portability, because Cogen goes hand in hand with, and is written in, COBOL.

For inquiries, write to:

bytel™ corporation

1029D Solano Avenue, Berkeley, CA 94706 (415) 527-1157 Telex: TLX 176609 BYTEL

CIRCLE NO. 204 ON READER SERVICE CARD

TABLE 1: *COBOL Compiler Features*

	DRI LEVEL II	IBM	MBP	MICRO FOCUS LEVEL II	MICRO FOCUS PRO.	MICRO-SOFT	REALIA	RYAN-MCFARLAND
VERSION TESTED	3.0	1.0	9.0	2.62	1.1	2.0	1.2	2.0B
PRICE	$700	$700	$1,000	$1,500	$2,995	$700	$995	$950
CERTIFICATION LEVEL	High	Low/int.	Low/int.	High	High	High[a]	—[b]	Low/int.
NATIVE/INT. CODE	Native	Int.	Native	Both	Native[c]	Int.	Native	Int.
LINK STEP	Optional	Yes	Yes	Optional	Yes[d]	No	Yes	No
STANDARD DOS LINK	No	Yes	Yes	No	No	Yes[e]	Yes	No[e]
DOS PATH SUPPORT	No	No	No	No	No	Yes	Yes	Yes
MAXIMUM CODE (KB)[f]	64/prog	64/prog	64/prog	64/prog	64/prog	64/prog	Unlim.	64 total
MAXIMUM DATA SIZE (KB)[g]	64	56	56	64	64	60	Unlim.	64
ALTERNATE INDEX KEYS	80	None	12	80	64	Unlim.	14	14
SORT	Yes	Extra	Yes[h]	Yes	Yes	Yes	No	No
FULL-SCREEN DISPLAY	Yes	Yes	Yes	Yes	Yes	Yes	No	No
FULL-SCREEN ACCEPT	No	Yes	Yes	Yes	Yes	Yes	No	No
INSERT/DELETE EDITING	—[i]	No	Yes	Yes	Yes	No	—[i]	No
PROGRAM EDITOR	No	No	No	No	Yes	No	Yes	No
SCREEN EDITOR	Yes	No	Yes	Extra	Yes	No	No	No
INTERACTIVE DEBUG	No	No	Yes	Extra	Yes	Yes	Yes	Yes
COMP. SIZE ON DISK (KB)	259	89	994	377	—[j]	130	342	64
MAXIMUM RUNTIME SYSTEM (KB)	88[k]	17	44	82[k]	0[l]	81	0[l]	37
MEMORY REQUIRED (KB)	192	64	192	256	256	192	192	128
MINIMUM NUMBER/DISKS[m]	3	2	5	3	3	2	3	1
LICENSING REQUIRED	No	Yes	—[n]	Yes	Yes	Yes[o]	No	Yes

[a] *Certified "with errors" for lack of full Communications and Debug Modules.*
[b] *Not certified; modules implemented place it in low-intermediate category.*
[c] *Intermediate code may be produced for testing.*
[d] *Link needed only for producing stand-alone applications, not for testing.*
[e] *Only assembly language subprograms may be linked to runtime system.*
[f] *64KB/program code limit applies to each separately compiled program, except Ryan McFarland.*
[g] *Data limit is total for main and all called programs in a program suite.*
[h] *Nonstandard Sort invoked through subprogram call.*
[i] *Insert/delete editing of screen fields may be programmed in COBOL.*
[j] *Combined with editor, debug, and other utilities. Bare compiler probably same as Level II.*
[k] *Runtime system not needed if compiler output is linked.*
[l] *No separate runtime system file is used.*
[m] *Working system with compiler, editor, debug, runtime, at least 180KB work space.*
[n] *Price includes license for 50 installations; unlimited license at additional cost.*
[o] *Agreement required, no charge.*

This table includes all eight compilers reviewed. IBM, mbp, and Micro Focus Professional COBOL are reviewed in this issue.

Like DOS and many of IBM's languages, this compiler is a variation of a Microsoft product (the product that evolved into Microsoft COBOL 2.0, reviewed in July). Source programs written for IBM COBOL are acceptable to the Microsoft 2.0 compiler. Features of IBM COBOL that are the same as Microsoft's (and therefore that were detailed in the previous review) are not described here. The compiler's output, although it is in standard DOS object format and must be linked, is really an intermediate code that is interpreted by a runtime system. System requirements and pricing for this and all of the compilers are provided in table 1.

mbp COBOL 9.0 is produced by mbp, a German company that distributes its products in North America through a California-based subsidiary, mbp Software and Systems Technology. This compiler also is certified at the low-intermediate level, but at the time when its peer products were under review this version was still in beta testing. The previous version was 7.4; version 8.0 was released only to the European market. Owners of version 7.x can upgrade for $250 (or less if they do so within 90 days of purchase).

mbp COBOL produces native machine code in standard DOS object files that must be linked before execution. A runtime system is required, but only to provide an interface to the operating system, not to interpret intermediate code. Although the documentation describes using floppy disks, this compiler is impractical without a hard disk.

IBM's manual is a model of conciseness. In barely 400 pages, it covers

operation of the compiler, linker, and indexed file rebuild utility, and provides an adequate language reference. No one could learn COBOL from it, but that is not its intent. Its only disadvantage is the same one suffered by the Microsoft 2.0 manual: certain language elements are arranged alphabetically instead of according to the order in which they appear in source programs.

mbp goes to the other extreme: its documentation is formidably hefty, consisting of two thick IBM-sized binders comprising a user's guide and a language reference. The language reference follows the format and wording of the ANSI standard, which makes for laborious reading at times. Although the manual is certainly complete, it is sometimes repetitious and in need of tighter editing. The user's guide provides all necessary information for operating the compiler and its ancillary utilities, but it could be clearer, especially the section on the screen editor.

The IBM compiler comes on two single-sided disks; as with most IBM products, it is preconfigured for PC hardware. A working system with adequate disk space for source and output files fits on two double-sided floppy disks; compiling, linking, and testing are performed with no disk swaps.

Because it predates DOS 2..x, IBM COBOL does not support directory path names in any form. An even worse consequence of its early vintage is the fact that COBOL programs refuse to run on systems with more than 544KB. This happens because memory switches on early PCs could not be set for more than 544KB, so early software reads only the low order bits of the memory size value, resulting in an out-of-memory condition on those systems in which the full memory size is given by more than four switches. The problem is solved by installing programs, such as print spoolers or RAM drives, that reside in high memory and so reduce the memory available to DOS below 544KB, or by running a memory utility that resets the upper limit to a lower value.

Although it is physically possible to run the mbp compiler on a dual-floppy-disk system, a hard disk is a virtual necessity. The compiler alone is close to one megabyte and takes up three full disks; a compile, link, and test cycle involves three or four disk swaps. The runtime system must be installed as a resident program; it takes up 44KB even when COBOL programs are not being run. Even on the 512KB or 640KB systems prevalent today, memory is not expendable, because RAM drives, print spoolers, keyboard macros, and other resident utilities use it up in a hurry. A transient runtime program, the kind used by most other compilers that require one, would make much more efficient use of system resources. Surprisingly for a newly updated product, mbp offers no support of DOS path names, either during compilation or execution.

mbp's handling of compiler error messages leaves much to be desired. Instead of being displayed on-screen, messages are placed in the listing file, which then must be examined with an editor. The only on-screen signal is the error count. The output can be directed to the screen, but the messages are more than 80 characters long and not easy to read on the fly. Even if the source listing is suppressed, the output still contains a multitude of messages besides errors, and offending lines are only referenced by number.

Fortunately the error messages are clear. One error-handling feature that seems promising is a compiler parameter that stops the compilation after a specified number of errors are discovered. The limit applies equally to warnings and fatal errors, however, so it is not really useful.

IBM error messages also do not display entire source lines, only a line

Slash Programming Time in Half!

With FirsTime™

- Fast program entry through single keystroke statement generators.
- Fast editing through syntax oriented cursor movements.
- Dramatically reduced debugging time through immediate syntax checking.
- Fast development through unique programmer oriented features.
- Automatic program formatter.

FirsTime is a true syntax directed editor.

FirsTime ensures the integrity of your programs by performing all editing tasks like moves, inserts and deletes along the syntactic elements of a program. For example, when you move an IF statement, FirsTime will move the corresponding THEN and ELSE clauses with it.

Even FirsTime's cursor movements are by syntax elements instead of characters. The cursor automatically skips over blank spaces and required keywords and goes directly to the next editable position.

FirsTime is a Syntax Checker

FirsTime checks the syntax of your program statements, and also:

- Semantics like undefined variables and mismatched statement types.
- The contents of include files and macro expansions.
- Statements for errors as they are entered and warns you immediately.

FirsTime is a Program Formatter

FirsTime automatically indents statements as they are entered, saving you from having to track indentation levels and count spaces.

FirsTime has Unique Features

No other editor offer these features:

The *Zoom command* gives you a top down view of your program logic.

The *View command* displays the contents of include files and macro expansions. This is valuable to sophisticated programmers writing complex code or to those updating unfamiliar programs.

FirsTime's *Transform command* lets you change a statement to another similar one with just two keystrokes. For example, you can instantly transform a FOR statement into a WHILE statement.

The *Move at Same Level command* moves the cursor up or down to the next statement at the same indentation level. This is very useful. For example, you can use it to locate the ELSE clause that corresponds to a given THEN clause or to traverse a program one procedure at a time.

FirsTime is Unparalleled

FirsTime is the most advanced syntax directed editor available. It makes programming faster, easier and more fun.

TO ORDER CALL (201) 741-8188

or write:

Spruce Technology Corporation

189 E. Bergen Place
Red Bank, NJ 07701

In Germany, Austria and Switzerland contact:

Markt & Technik Software Verlag
Munchen, W. Germany
(089) 4613-0

FirsTime for **Turbo Pascal $74.95**

FirsTime for **MicroSoft & IBM Pascal $245**

FirsTime for **C $295**

TABLE 2: *Compilation Speed and Code Size*

	DRI LEVEL II	IBM	MBP	MICRO FOCUS INT.	MICRO FOCUS NATIVE	MICROSOFT	REALIA	RYAN-MCFARLAND
ERATOSTHENES SIEVE (80 lines)								
Compile	2:08	0:44	3:44	0:30	1:46	1:04	0:49	0:51
Link	—	0:41	0:27	—	0:21	—	0:29	—
Code size (KB)	10.2	16.3	20.1	10.0	58.7[a]	10.2	15.0[a]	1.5
FILE I/O (350 lines)								
Compile	3:37	1:59	7:47	0:40	2:37	1:52	1:10	2:34
Link	—	1:06	0:42	—	0:20	—	1:02	—
Code size (KB)	8.4	24.7	19.4	4.1	57.6[a]	5.8	25.8[a]	3.1
MEDIUM PROGRAM (500 lines)								
Compile	5:20	3:56	10:49	0:50	3:45	3:16	1:15	3:27
Link	—	0:48	0:50	—	0:24	—	0:28	—
Code size (KB)	10.6	13.8	20.7	10.6	57.4[a]	7.5	13.4[a]	3.6
LARGE PROGRAM (1,000 lines)								
Compile	9:33	8:59[b]	—[c]	1:25	7:40	6:43	1:43	7:05
Link	—	—	1:22	—	1:41	—	0:37	—
Code size (KB)	18.6	—	30.3	18.6	66.2[a]	14.7	21.4[a]	6.4
GIBSON MIX (1,500 lines)								
Compile	9:34	8:57	—[c]	1:49	6:51	6:35	2:03	—[d]
Link	—	1:17	1:08	—	1:28	—	1:07	—
Code Size (KB)	24.6	34.4	34.4	15.9	74.1[a]	21.3	36.7[a]	—

Execution times are in minutes:seconds.

[a] *These code files are self-contained; all others need a runtime system to execute.*
[b] *Produced spurious errors on compilation; it was not linked.*
[c] *Could not compile on floppy disks because work files exceed 400KB.*
[d] *Would not compile because program included features not supported by this compiler.*

All compilation tests were run on floppy disk systems. For both Micro Focus compilers (Level II and Professional), compilation times are essentially the same and code sizes are identical. Times shown for Micro Focus native code are the sum of the times for the intermediate and final compilation steps.

number and the element in error. As in Microsoft COBOL, many errors are followed by "unrecognized element" messages for every remaining item in the statement. A simple "rest of statement ignored" would be sufficient.

For linking, both compilers use the standard DOS LINK program. IBM provides a copy with the compiler; mbp provides a utility that builds command files for PLINK86, a linker that allows greater control over the overlay structure of the finished application. For both, compilation and link steps are easily automated with batch files.

PERFORMANCE CONTINUED

Compilation times and output code sizes are listed in table 2. mbp COBOL turns in the slowest compile times of the eight compilers, partly because of the large size of the overlays that must be read in and the huge work files that are created. Programs of 1,000 and 1,500 lines could not be compiled on the test system because the work files exceed the capacity of a floppy disk.

For the Gibson program, compiled on a PC/XT, the two work files totaled over 600KB. The compilation times for the two largest programs are not listed because they were obtained on a hard disk system and therefore are not directly comparable to times on a floppy-disk-based system. When all compilers are compared on a hard disk system, mbp's compile time still puts it last.

IBM COBOL seems to have problems compiling large programs. The 1,000-line program would not compile cleanly; it generated nonsense error messages for some source lines num-

bered above 950. This must be entirely because of the size, not the syntax requirements of the compiler, because this test program consists of repeated groups of identical statements with different paragraph names. If most of the program can pass compilation, there should be no reason why the same statements should suddenly become unacceptable only because of their position in the source. If the compiler simply runs out of work space, it should issue a message and terminate, not send the programmer on a wild goose chase after nonexistent errors.

The Gibson Mix program, 1,500 lines long, also produced nonsense errors for lines near the end, but after several rearrangements of comment lines, it finally compiled cleanly. This is a classic benchmark developed by IBM in the 1960s for testing COBOL on mainframe systems. It consists of a series of internal operations most typical of COBOL applications, and one of the results it produces is a *calculated S-profile*, a weighted average of the times needed to perform one iteration through each of the operations. The weighting goes according to the estimated frequency of each operation in a typical application.

Results for performance benchmarks for internal operations are shown in table 3. IBM is second to last; only Microsoft is slower. mbp performs respectably (third overall), falling behind Realia (see "COBOL Performs," Ted Mirecki, June 1985, p. 58) and Micro Focus native code.

The results for file I/O performance in table 4 show little variation among the compilers in sequential and relative I/O, but indexed I/O and sorting show significant differences. mbp is the slowest in indexed I/O, but improves significantly if the number of buffers specified in the CONFIG.SYS file is increased above the default. It is the fastest in the sort; however, its sort feature is implemented by subprogram calls, not as the standard COBOL Sort/Merge module defined by ANSI. IBM COBOL does not implement the sort, but an additional program provides it (discussed below under Sort/Merge).

In indexed I/O, times for IBM COBOL are not directly comparable to the others because that compiler does not support alternate index keys. With only prime keys in the index structure, reading and writing are faster than with a multiple-key index. When compared with the others in single-key indexed operations, IBM performs well, coming in third after Realia and Microsoft.

TABLE 3: *Benchmark Results—Internal Operations*

	DRI LEVEL II	IBM	MBP	MICRO FOCUS INT.	MICRO FOCUS NATIVE	MICROSOFT	REALIA	RYAN-MCFARLAND
ERATOSTHENES SIEVE (per iteration)								
Binary	0:30	5:53	0:15	2:04	0:30	16:02	0:03	3:40
Decimal	1:16	5:44	2:09	2:29	1:16	18:08	0:18	3:47
DECIMAL ARITHMETIC								
5,000 iterations	0:34	1:37	0:34	1:04	0:34	4:04	0:10	1:10
CHARACTER OPERATIONS								
500 iterations	0:40	4:37	1:15	2:33	0:40	13:10	0:10	3:32
SCREEN DISPLAY								
10 iterations	0:47	1:00	0:05	0:06	0:06	0:30	0:35	0:36
GIBSON MIX (10,000 iterations)								
Total time	17:27	2hr 34min	33:40	1hr 18min	17:26	6hr 21min	5:00	—[a]
S-profile (seconds)	11.0	97.4	16.3	63.7	11.0	389	4.5	—[a]

Execution times in minutes:seconds unless otherwise noted.

[a]*Could not run the Gibson program because some of the features are not implemented by this compiler.*

This table demonstrates the performance of compiled code in internal operations. Results for the Micro Focus Level II and Professional compilers are the same.

Memory utilization for IBM and mbp is the same and typical of the medium memory model implemented by most of these compilers: 64KB of code per separately compiled program, 56KB of data for the entire application, with total code size limited only by available memory. Those compilers that limit data to less than 64KB allocate a full segment but use some for overhead.

Two bugs were discovered in the code produced by the IBM compiler: first, the Julian date routine (days since beginning of year) returns the wrong result in leap years and second, during full-screen accepts, the cursor sometimes disappears (refer to the Console I/O section below).

LANGUAGE IMPLEMENTATION

IBM COBOL provides the lowest level of language implementation of all the compilers. The following list is a summary of unsupported features and other restrictions:

- Variable length tables (OCCURS DEPENDING ON) are not supported.
- Occurrences of data items in tables are limited to a maximum of 1,023.
- RENAMES clause in the Data Division is not supported.
- CORRESPONDING phrase for MOVE, ADD, and SUBTRACT is not supported.
- Division produces no remainders.
- Alternate index keys are not supported.
- Interprogram Communication, Library, and Segmentation modules are implemented to Level 1 standards only.

mpb provides Level 2 implementation of the Nucleus, Table Handling, Segmentation, and the three I/O modules and Level 1 implementation of Library and Interprogram Communication, as detailed below.

Table 5 shows the implementation of computational data types. IBM is the same as Microsoft, and mbp and Ryan-McFarland are similar, except for COMP-6 (unsigned decimal) which is implemented only in RMCOBOL. Picture clauses should not be specified for mbp binary (COMP-1) items; if they are, the compiler issues warning messages that count toward any error limit specified as a compiler parameter and may prematurely end the compilation.

Console I/O. IBM COBOL implements full-screen I/O in the same way as Microsoft does—by a SCREEN SECTION extension to the Data Division. In addi-

tion to the same poor user interface (refer to the July review of Microsoft), the IBM version exhibits one other problem: when the compiler is running on a color/graphics adapter, the cursor randomly disappears in successive executions of the same program.

mbp COBOL provides two extended screen interfaces. One is the same as Ryan-McFarland's (see "COBOL Performs," June 1985, p. 58). The fields of a screen form are specified as multiple operands in one DISPLAY or ACCEPT statement, with screen coordinates and video attributes specified for each field. This allows some measure of compatibility with the many RMCOBOL applications currently on the market.

In addition, mbp provides a true full-screen interface accessible through calls to a Screen Management System. The applications program may display a whole screen and accept input in its variable fields with a single subprogram call. A stand-alone screen editor is provided for laying out screen forms and defining the usage and video attributes of each of its fields. This system is straightforward and fairly easy to use, although its poorly organized, opaque documentation suggests otherwise.

File I/O. Of the eight compilers tested, IBM's is the only one that does not support alternate index keys. Otherwise, its file system is the same as Microsoft's. One unusual aspect of that system is that assignment of program file names to DOS files is done in the FD (file definition) statement of the Data Division, not in the SELECT statement of the Environment Division as it is in most other implementations of COBOL.

mbp line sequential files are declared not by the phrase ORGANIZATION LINE SEQUENTIAL, but by appending a file type letter to the DOS file name. In both sequential and relative files, each record is preceded by a length word, and relative files also have a record number zero that serves as a file header containing the physical record length and the number of records in the file. Indexed files contain both keys and data in the same file; the structure is not documented, but a stand-alone utility is provided for recovering files with damaged indexes.

Both IBM and mbp allow either static or dynamic file assignment in the usual way, by assigning to a file name either a literal string or an identifier that must contain a valid DOS file name at open time. mbp programs are unique among these compilers in that if at run-time the file name is invalid, the user is prompted to enter another name.

TABLE 4: *Benchmark Results—File Operations*

	DRI LEVEL II	IBM	MBP	MICRO FOCUS	MICROSOFT	REALIA	RYAN-MCFARLAND
SEQUENTIAL I/O							
100 records							
Write	0:05	0:06	0:07	0:05	0:05	0:04	0:05
Read	0:04	0:04	0:04	0:04	0:04	0:03	0:04
Copy	0:10	0:13	0:10	0:10	0:09	0:06	0:16
300 records							
Write	0:14	0:16	0:16	0:14	0:16	0:08	0:15
Read	0:13	0:13	0:13	0:13	0:14	0:08	0:13
Copy	0:24	0:28	0:48	0:24	0:27	0:15	0:27
RELATIVE I/O							
100 records							
Write[a]	0:05	0:06	0:07	0:05	0:07	0:05	0:05
Read[b]	0:24	0:06	0:24	0:24	0:24	0:24	0:24
Update[c]	0:55	0:15	0:56	0:55	0:55	0:54	0:53
300 records							
Write	0:14	0:15	0:15	0:14	0:17	0:14	0:15
Read	0:43	0:13	0:45	0:43	1:12	0:49	1:06
Update	2:17	0:44	2:18	2:17	2:17	2:13	2:11
INDEXED I/O							
100 records							
Write[d]	1:12	0:30[g]	2:40	2:10	0:30	0:17	4:06
Read[e]	0:25	0:10	1:18	0:24	0:22	0:12	0:48
Update[f]	—[h]	—[i]	7:20	3:01	1:05	0:24	4:27
300 records							
Write	4:48	1:35[g]	11:12	6:19	1:35	2:59	6:26
Read	1:12	0:37	3:59	1:19	1:10	1:16	2:14
Update	—[h]	—[i]	31:18	9:17	6:22	3:52	9:34
SORT							
300 records	2:27	—[j]	0:08[k]	1:41	0:29	—[j]	—[j]
1,000 records	13:52	—[j]	1:03	8:37	1:59	—[j]	—[j]

Time in minutes:seconds.

[a] *Write nonconsecutive record numbers.*
[b] *Read nonconsecutive records, in different order than written.*
[c] *Read nonconsecutive records, in different order than written, rewrite same record.*
[d] *Write not in key sequence, prime key and one alternate key.*
[e] *Read in different sequence than written.*
[f] *Read in different sequence, change alternate key and rewrite.*
[g] *Does not implement alternate keys; times are not directly comparable to the others.*
[h] *Could not update alternate key correctly.*
[i] *Test not performed because alternate keys not implemented.*
[j] *Not applicable. Sort feature not implemented.*
[k] *Nonstandard sort routine called as a subprogram.*

I/O tests were performed on floppy disks, with the default number of disk buffers.

Table Handling. mbp provides standard Level 2 table handling facilities; IBM provides most of Level 2 except variable length tables (OCCURS DEPENDING ON). IBM COBOL further restricts tables by limiting them to no more than 1,023 occurrences, but this is only an inconvenience because a large table can be defined as several contiguous smaller ones. The Sieve program, which requires a table of 8,191 items, was compiled and run without difficulty by defining the table in nine pieces. Because the runtime system does not

What to do while your COBOL programs compile and execute:

1. Wait.
2. Wait some more.
3. Stop waiting. Call Realia.

Patience isn't always a virtue.
Realia COBOL is fast:

Compilation Speed (minutes:seconds)

Lines in Program	Realia COBOL	mbp COBOL	Level II COBOL	R-M COBOL	Microsoft COBOL
1,000	:51	8:33	3:42	5:05	5:11
5,000	3:30	48:07	16:58	*	45:26

*Could not successfully compile the program.

Execution Time Ratio
(Gibson Mix; calculated S-Profile)

Realia COBOL	mbp COBOL	Level II COBOL	R-M COBOL	Microsoft COBOL
1.0	3.6	14.7	21.6	22.3

All benchmark tests were performed on an IBM PC-XT with 192KB of memory. IBM PC-XT is a registered trademark of International Business Machines Corporation; mbp COBOL, of mbp Software and System Technology; Level II COBOL, of Micro Focus; R-M COBOL, of Ryan-McFarland; and Microsoft COBOL, of Microsoft.

Sieve of Eratosthenes

0.818 seconds per iteration

Realia COBOL is written in COBOL. We offer you the tools we use ourselves:

- Our FOLLOW-THE-SOURCE™ interactive symbolic debugger. Works with normal native code.
- A speedy full-screen editor that handles very large files.
- Mainframe IBM VS COBOL compatibility.
- Interfaces to Assembler and C.
- No royalty or run-time fee.
- No limit on program size, up to available memory.
- In our new release, no need to insert the product diskette when you're using a hard disk.

Realia COBOL costs $995. Qualified companies can try it for free. Call us. And ask about our products Spacemaker™ and Termulator™. Visa/MC/AmEx.

What are you waiting for?

REALIA inc.
10 South Riverside Plaza
Chicago, Illinois 60606
(312) 346-0642
TELEX: 332979 (REALIA INC)

TABLE 5: *Computational Data Types*

	DRI LEVEL II	IBM	MBP	MICRO FOCUS	MICROSOFT	REALIA	RYAN-MCFARLAND
COMP	Binary integer, 1-8 bytes	Display	Unpacked decimal	Binary integer, 1-8 bytes	Display	—[a]	Unpacked decimal
COMP-0	—	Binary integer, 2 bytes	—	—	Binary integer, 2 bytes	—	—
COMP-1	—	—	Binary integer, 2 bytes	—	—	—	Binary integer, 2 bytes
COMP-3	Packed decimal	Packed decimal	Packed decimal	Packed decimal	Packed decimal	Packed decimal	Packed decimal
COMP-4	—	—	—	—	—	Binary integer, 2-4 bytes, high-low	—
COMP-5	—	—	—	—	—	Binary integer, 2-4 bytes, low-high	—
COMP-6	—	—	—	—	—	—	Unsigned packed decimal

[a]*May be defined as any other usage, including display. Default is COMP-4.*

Different meanings for the various COMPUTATIONAL usages is one obstacle to source code portability among these COBOL compilers.

check subscripts against table bounds, a subscript greater than 1,023 can be used to refer to any item within the concatenation of the smaller tables. Not very elegant, but it works.

Library. Both compilers provide basic Level 1 support of the COPY statement: inclusion of source files into the compiled program, with no capability of text substitution. mbp accepts a library name, but treats it as documentation only. In both compilers, the copy file is read in from the default drive unless a drive designation is coded in the COPY statement; most of the other compilers look for copy files on the same drive as the main source program.

For mbp, the default drive is dedicated to the compiler, so the drive containing the copy file must be explicitly coded into the source program. The source code must be changed every time the disk containing the copy file is put in a different drive. The IBM compiler is usually run with the source disk in the default drive, so copy files that do not include explicit drive designators are read from whichever drive contains the source disk.

Debug. IBM's debug facility is the same as Microsoft's. Of the standard ANSI Debug module, only debugging lines are available, but the TRACE and EXHIBIT extensions are provided. Unlike Microsoft, however, IBM has no interactive debugger, and the standard DOS DEBUG is next to worthless because the runtime system effectively isolates it from the compiled code. For all practical purposes, therefore, interactive debugging is not available with the compiler from IBM.

mbp COBOL has full ANSI Debug facilities, including USE FOR DEBUGGING procedures in the Declaratives Section and the predefined data name DEBUG-ITEM. (These facilities were explained in the previous article.) In addition, mbp provides an interactive debugger with the usual repertoire of commands for tracing, single-stepping, setting and clearing breakpoints (up to 20), displaying, and modifying data. A source listing is required because Procedure Division code is referenced by line number, but data items are referenced by their source names, so an address map is unnecessary.

HIKE-UP YOUR AT!

WHY WAIT? GET 100% COMPATIBLE HARD DISK AND TAPE BACKUP FOR YOUR IBM® PC AT™ FROM MOUNTAIN TODAY.

Hike-up your basic AT to an enhanced AT with Mountain's internal hard disk. Installs in minutes using the AT's disk controller, power supply, and DOS 3.0. Choose from 20MB, 35MB or 120MB (formatted).

And don't forget our AT internal tape that lets you back up your irreplaceable files with complete confidence. It's fast. Economical. Easy-to-use. And installs in minutes. Complete with menu-driven software. Choose from 27MB or 60MB (formatted).

Both come network-ready to support Novell, Orchid PCnet,® NESTAR and 3COM Ether Series.

Why wait for IBM's hard disk when you can get just the capacity you need today. Plus reliable tape backup.

See them today at Entre (1-800-HI-ENTRE), Morris Decision Systems, MBI Business Centers, Office Technology Plus (Federal Agencies only), and other leading computer stores.

For more information, write: Mountain Computer, Inc., 300 El Pueblo Road, Scotts Valley, CA 95066. Or call (408) 438-6650. TWX 910-598-4504.

Mountain®

THE PEAK OF PERFORMANCE

1-800-458-0300 (in California, 1-800-821-6066)

In Canada, call Parity Plus Inc. (416) 673-3321.
In France, call Compusol 01-530 07 37.
In Sweden, call Nordic Data 08-730 50 50.

CIRCLE NO. 154 ON READER SERVICE CARD

Computational data is displayed or modified in hexadecimal form, with no conversion to or from character representation. Modification of program flow is possible, but execution may be transferred only to a procedure name, not to an arbitrary line number. In general, the mbp debugger is comparable to Microsoft's; it operates at a higher level than Ryan-McFarland's, but is not as convenient as those from Realia and Micro Focus, which display the source code as it is executed.

Inter-Program Communication. Subprogram calls are handled at ANSI Level 1 by both compilers, that is, all subprograms must have their names hard-coded as literals in the CALL statements and be linked into one load module with the main COBOL program. mbp COBOL, however, has available the functional equivalent of dynamic subprogram loading through the chain facility (described below).

mbp provides sketchy but adequate documentation for calling assembly language routines. The linkage conventions are somewhat complex, because a parameter count and length of each parameter are passed on the stack along with the parameter addresses. Even though there is only one data segment, each parameter address is in far format, that is, includes both offset and segment components. Despite these anomalies, mbp claims that "there should be no major difficulty" in calling C and Pascal, but provides no instructions on how to structure such called programs. Interfacing to these other languages requires much care in matching the memory model, specifying the correct number or parameters (different than the number in the calling COBOL program), and the type (value or pointer) of each.

IBM's assembler documentation is quite complete, and the conventions conform to the medium memory model: a far return address and near (offset only) parameter addresses are passed on the stack. However, the example given (an assembly language routine to add two binary integers) is in error because it fails to point out that IBM COBOL stores 16-bit integers with the high order byte at a lower address, whereas assembly language code assumes the reverse. This example works only as long as both values and the sum each fit within one byte.

Besides calls, both compilers allow chaining to other programs. IBM's CHAIN verb, an extension to the language, reads in a new main COBOL program to replace the current one. Data may be passed to the new program in much the same way it is passed to a subprogram.

mbp's implementation of chaining is one of its best features. This compiler supports chaining by means of calls to several chain routines, which then read in the chained COBOL program. Depending on the routine used, the program issuing the chain may be replaced or remain in memory and subsequently regain control at the point following the chain call. This provides the functional equivalent of dynamically loaded subprograms, and the further advantage that the program to which the routines are chained is a main program with its own data segment.

Chaining is controlled by a special COBOL program called a chain controller, provided in source form so that it may be customized. Besides chaining COBOL programs, it may also execute other programs, including DOS utilities, passing them command line arguments.

Sort/Merge. mbp's Sort/Merge capability is implemented using subprogram calls, not the standard, ANSI-defined SORT and MERGE. The link library provides two sort programs. One, called mbpSORT, performs the same function as a standard COBOL SORT statement with USING and GIVING phrases. It reads in a specified file, sorts it, and writes it out to a specified output file; it offers no capability of processing or selecting the records before or after the sort. Input and output files may be any COBOL-supported types. mbpSORT also is provided as a stand-alone program executable from DOS.

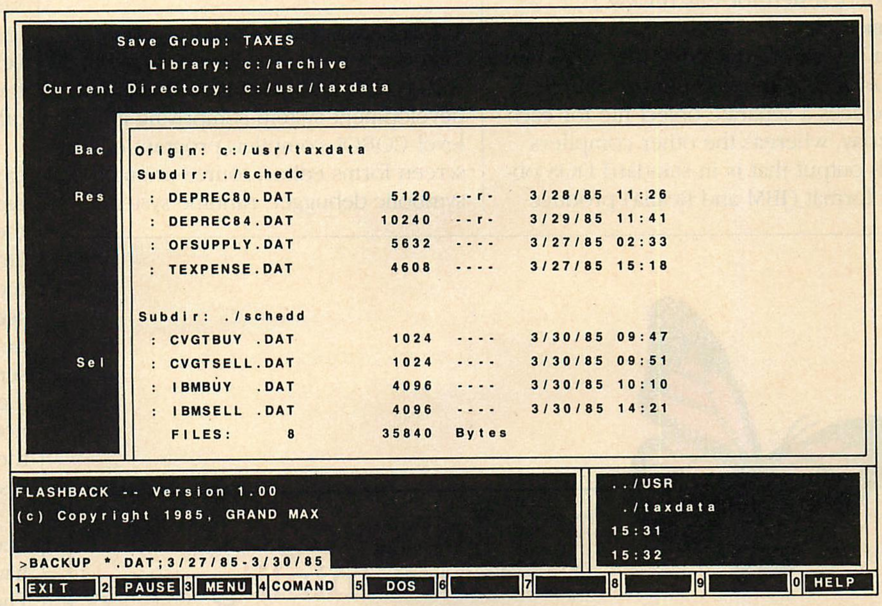
The other sort routine, called RSORT, provides the same capabilities as a SORT statement with input and output procedures. The input and output procedures are written in one subprogram, most conveniently in COBOL, which is called by RSORT to open the input file, read or select the records to be sorted, write them out after sorting, and close the output file.

As shown in table 4, mbp's sort performance is the best, probably because of the large memory work areas reserved for sorting. The memory usage may be reduced, at the cost or slower sort performance, by linking in one of several object files that reserves less than the default work space in the load module. This is a good implementation of Sort/Merge, but it could be improved by providing these capabilities through standard COBOL syntax.

IBM COBOL does not implement the Sort/Merge module, but another IBM/Microsoft product, PC-Sort, can add this capability. Besides a stand-alone,

FILE MANAGEMENT is *Fast Backup* and much, much more...

FILEMASTER supports a versatile file selection system which lets you process files by name key, modification date range, or name key range, using relative or absolute path specifications. . .



FILEMASTER

The definitive backup and file management system for hard disk users.

FILEMASTER is a complete set of tools for managing the complexities of the DOS file system. In one integrated package, you get seven powerful file management utilities:

FLASHBACK; over twice as fast as the standard DOS backup utility, with options for data compression, file encryption, data validation, and on-line directories of file archives stored on diskettes. Once a save set is created, FLASHBACK will automatically save only those files which have been changed or added since the previous save.

Using sophisticated data compression techniques, FLASHBACK saves storage space for file backups, and lets you create compact, encrypted data packets for use in computer-to-computer data transfers.

BDIR; gives you a quick directory listing of files stored off-line in your backup archives.

FINDEX; an application driver which lets you select and process multiple file sets anywhere on your disk performing any of several standard functions, or linking with applications of your own. Once FINDEX starts, it works automatically, crossing data paths without user intervention.

TEXTED; a global string search & display, search & modify utility which allows you to make large scale changes to files on your disk, both binary and ascii.

TUNEUP; a hard disk tune-up process which sorts directories, frees up wasted directory space, and optimizes disk storage to make your hard disk run smoother and faster.

DISCOPE; a powerful storage analysis program which allows you to examine, compare, and modify disk sectors, either as part of a file chain, or in arbitrary combinations.

EXPATH; a path extension utility which allows you to reference files in different directories by name without changing your current directory.

You've probably seen similar utilities advertised individually before, and at prices well over $100 each. With FILEMASTER, you get them all for only:

$49.95

including a complete user manual & binder, contemporary multi-window, pop-up menu interface, and a support line for technical assistance.

Hardware Requirements: IBM-PC XT, Compaq-Plus, or compatible equipment; DOS version 2.xx; minimum 128K, 256K recommended for optimum performance.

IBM & IBM-PC are trademarks of IBM Corporation.
Compaq-Plus is a trademark of the Compaq Computer Corporation.

FILEMASTER is available only from Grand Max Software Corporation. To order with VISA or Mastercard, dial:

1-800-624-3285

In Oregon: (503) 635-8295

Grand Max
SOFTWARE
CORPORATION

68 LEONARD STREET • LAKE OSWEGO, OR 97034

DOS-level Sort/Merge program, PC-Sort provides a means to patch the IBM COBOL compiler to allow it to recognize the standard sort and merge statements; a link library of the necessary routines is also included. (This product was not tested for this review.)

Segmentation. Standard ANSI segmentation is supported at Level 1 by IBM and at Level 2 by mbp. The mbp compiler produces a separate object file for each overlay, whereas the other compilers with output that is in standard DOS object format (IBM and Realia) produce only one object file. mbp's method allows more flexibility in structuring the load module with PLINK86, but it is inconvenient to use with the standard DOS linker because all of the object files must be specified as input to LINK.

PROFESSIONAL COBOL

This is a very impressive product, but at $3,000 it should be. It is an integrated development system comprising a high-level COBOL compiler, program editor, screen forms editor, source-oriented symbolic debugger, runtime system, linker, and library building utility. The compiler at the heart of Professional COBOL is essentially the same Level II compiler reviewed in detail in the July issue, and its performance in compilation and execution speeds was the same. This review concentrates on the features that are different from the Level II implementation, and on the interaction of the compiler with the other components of the system.

The manuals are improved slightly from the Level II versions. A larger typeface and printing on matte paper in the standard 5½-by-8½ size help, but larger type on smaller pages means more pages, and the ringed binders seem crowded. Information is easier to find because the table of contents is sensibly summarized and each page has a chapter header. The index tabs are not as useful as they might be because they are labeled with chapter numbers instead of subjects.

The language reference has not been changed; it still follows the exhaustive, not-very-readable ANSI format. The operating guide has been extensively rewritten to cover the operation of the utility modules, and all Level II updates have been incorporated. It is incomplete in its treatment of the utilities: some of their capabilities are discovered only through experimentation or by studying the example programs. Given the size and scope of this system, a complete user's guide would be a sizable volume; instead, the user interface, with its comprehensive on-line help, makes the system easier to learn by doing. Overall, the documentation and examples are more than sufficient to get an experienced programmer started.

The system requires three disks for its operation. Two are system disks containing the compiler and utilities; the third holds the help library and space for source and output files. Micro Focus supplies two copies of each of the system disks, even though they are not copy-protected. This permits immediate use, even though users may still want to make working copies and keep both distribution copies for backup.

Despite its size (800KB of files), Professional COBOL is practical even on a dual-floppy-disk system. The disk with the help and source files is kept in the B: drive and one of the system disks is placed in the A: drive; prompts are issued when system disks need to be swapped. A hard disk is much more convenient. However, the real story on ease of use is the integration of all the functions: editing, compiling, testing, and linking are performed without once

seeing the DOS prompt. Upon starting the system, the user is greeted by the following main menu choices:

Edit. This provides access to the program editor and the screen editor.

Check. This is the first compilation step, producing intermediate code for test runs or for debugging.

Animate. The interactive debugger.

Compile. This is the second compilation step, producing native code.

Run. This allows the user to execute either intermediate or native code.

Library. This is used to combine source or code files into libraries.

Build. This is used to link intermediate or native code files to the runtime system for stand-alone applications.

All of these modules use the function keys to navigate through a hierarchy of menus; the command structure is consistent throughout. A menu of function keys is always displayed at the bottom of the screen, and pressing Ctrl or Alt changes the menu to reflect the functions they make available. Pressing Esc exits the user from a module to the main menu or moves him to the next higher menu within a module. At any point the F1 key brings up a help screen that explains the functions on the current menu. File name specifications may be entered on a command line, or a directory list may be requested and a name chosen by pointing to it with the cursor keys. Screen response is practically instantaneous in all of these operations.

The separation from DOS is not as complete as Micro Focus claims. Some common operations, such as deleting, renaming, and copying files, are not provided within the COBOL environment, but it is not necessary to exit to DOS to perform them. A second copy of COMMAND.COM may be started from within COBOL; it displays the familiar DOS prompt and the user may perform any internal command or run any executable program for which there is sufficient memory. When done, the EXIT command returns the user to the point within COBOL from which DOS was invoked. This requires that the command interpreter and any external command files be on-line. On a floppy-disk-based system, these are most conveniently placed in a RAM drive and a SET COMSPEC=C:/COMMAND.COM command issued before entering COBOL. Unlike DOS, Professional COBOL has no trouble finding COMMAND.COM where COMSPEC says it is.

Although the system is designed primarily for interactive use during program development, batch operations can automate the production of finished applications at the end of the development cycle. Each of the main functions may be invoked directly from DOS without going through the main menu, permitting convenient batch control of several functions in sequence.

Overall, the whole system is a joy to use. But the perfect software product has yet to be made, and as good as Professional COBOL is, it could be better. Directory support is only slightly better than in Level II COBOL. The default drive and its current directory may be changed, but multiple directories are not supported: all files are read from the current directory. In the interactive mode, files are specified by name only, without extensions, because each module has a default extension that cannot be overridden. For those functions that operate on more than one type of file (for example, the Run module may execute either intermediate or native code), the file must be selected from a directory list because distinctions by extension cannot be made on the command line. This unnecessarily slows

TAPE IT EASY.

Boy, are you in for a surprise. You know you need mass storage. And you know you need some kind of backup.

But what you don't know is that making a backup from your hard disk can take the patience of a saint. And tax the memory of an elephant.

Unless, of course, you have a little TLC. The new TrimLine Combo from Corvus. You get mass storage (20 Mbytes). The incredible speed and efficiency of a hard disk. And an integral tape backup.

But there the similarity between a Corvus and anything else ends.

First of all, your TLC fits in beautifully. Between your IBM PC's monitor and base. No bulky boxes. And no clumsy cables and wires to trip over.

Next, you get the world's easiest tape backup system. If you like using command lines, fine. The TLC will accommodate you. But if you're more interested in simplicity, just use one of the TLC's menus. And one of your fingers.

You can back up the entire disk. All your data for the day. A group of selected files. Or just one single file. As a matter of fact, you can back up whatever you want. Which will save you quite a bit of time and aggravation.

You'll also find backed up files in seconds with the TLC's Directory. No more searching an entire tape just to find one file.

Finally, you'll enjoy the backup speed of the TLC. There's no formatting of blank tapes (which can save you about two hours). And no worry about losing data to bad sectors the format has called good.

Scared? Don't be. Just tape it easy. Check out the TrimLine Combo from Corvus.

In addition to all the features, it's one of the lowest-priced tape backup systems you can find.

For more information and the name of your nearest Corvus TLC dealer, call (800-4-CORVUS).

CORVUS

CIRCLE NO. 132 ON READER SERVICE CARD

down the operation when the user knows which file he wants.

In addition, the DOS interface is invoked by the awkward keyboard combination of Ctrl-Shift-Break, which is dangerously similar to both the Ctrl-Break and Ctrl-Alt-Del combinations. Furthermore, documentation of the interface is buried in a chapter called "Advanced Features" in the user's guide. Given the level of expertise required to develop the kinds of applications for which this system is obviously intended, Micro Focus' treatment of operating system functions as "advanced" seems strange, as do its contortions to protect the user from DOS.

The Program Editor. This feature makes intelligent use of the PC's cursor control and function keys. The function key menu at the bottom of the screen runs together with the last few lines of the displayed text, but its continuous presence is a reassuring guide through the many capabilities of this feature-laden editor. Cursor and text moves are acceptably rapid, and scrolling is lightning-fast and free from interference even on a color/graphics adapter.

File operations are quick. When a file is loaded, the first screen is displayed almost immediately, then the rest of the file is read from disk; this significantly improves the perceived response time. The complement of editing functions includes the usual ones: insert or overtype mode, block deletes and moves within and between files, search, and replace. The intelligent design of the user interface and the ever-present menus make learning easy. Unfortunately, the editor does not support macros or multiple windows.

Word wrap, adjustable tabs, line split, and line join functions permit limited text processing, such as writing documentation, but this editor is primarily tailored to the format of COBOL source programs. One very useful feature is multilevel editing of copy files or called programs. When the cursor is positioned on a COPY or CALL statement, or any character literal for which the value could be a file name, a single keystroke brings up the named file for editing, without ending the edit of the current file. Terminating the edit at the lower level returns to the higher one. The limit of this nesting is not specified in the documentation, but by actual test, five levels of called programs can be edited. If this editor does not fully measure up to the best stand-alone program editors, its integration with the forms editor, debugger, and runtime system more than compensates.

The forms editor is used to lay out forms for full-screen display and accept operations. It is entered from the program editor, after positioning the cursor at the point in the Data Division where the data definitions of the screen fields belong. The screen form is designed by typing the constant prompt fields and entering COBOL pictures for the data fields. Drawing enclosing boxes is easy because the keyboard supplies the required character set, and video attributes, including color, are painted with cursor moves. In addition, the function key menu may be turned off in order to avail the forms editor of all 25 screen lines for the form.

After the completed screen layout is saved to a file for subsequent recall and modification, it must be converted to a set of data definitions for copying into the program that uses the screen. A single command generates a file of data definition statements, inserts the required COPY statement into the program, and returns the user to the program editor. The names of the generated screen fields are not mnemonic-oriented; they are composed of the screen name and the row-column coordinates. However, the copy file may be edited to change the names to something more meaningful. The hierarchi-

cal editing capability described above makes this very easy.

To generate the code that actually performs the screen I/O, the user points to the place in the Procedure Division where the I/O is to occur, returns to the forms editor, and executes the code generation command. Procedure Division statements are placed directly into the program, not into a COPY file as are the data definition statements. Once back in the program editor, the user may modify these statements or insert other statements among them as required.

This screen generation system is highly functional and easy to use. Of the three such systems that were tested (the others are from Digital Research and mbp), this is the only one that automates the updating of the program with every change to the screen form. With the other systems, the COBOL program must be separately updated after every change to the length or order of the screen fields. Professional COBOL's forms editor has one minor inconvenience: each time a copy file is generated, a new COPY statement is inserted into the program, so with each update to the layout, the extra statement must be deleted, otherwise duplicate data definitions will result.

Compilation requires two steps: first, a syntax check reports errors and produces intermediate code for debugging purposes, then the compile proper converts the intermediate code to native machine code for more efficient execution. In the Level II compiler, these two steps were called compilation and generation, respectively. During the syntax check, the default is to send the listing of the source program to the screen, but it may be sent to a file or to the printer, or suppressed entirely. Leaving the default in effect is convenient overall; for programs of moderate size, the checking speed is not noticeably degraded by the display. When the screen listing is suppressed, only lines with errors are displayed. At every source code error, the process pauses and the user is queried whether to continue. A negative answer puts him in the editor with the source file loaded and the cursor on the offending line, ready for editing. This pause-and-query feature may be turned off to produce a list of all the error messages for subsequent correction. Unlike in the Level II compiler, in this compiler, errors involving duplicate data names do not cause the checker to enter an endless loop.

The second compilation step converts the intermediate code to native machine code. All source code errors must be corrected during the syntax checking phase because the second step cannot report errors. Once the program is clean, this step runs uneventfully.

The interactive debugger, for reasons which soon become apparent, is called Animator. An excellent debugging tool, this is probably the best of the many fine features of Professional COBOL, one of the finest program development tools for any language.

Animator operates on the intermediate code and symbol tables produced by the syntax check. Upon entry to the debugger, most of the screen displays the source program, beginning with the first Procedure Division statement past the Declaratives Section. Execution may be controlled in several ways: single stepping statement by statement, setting breakpoints at a source line to which the cursor is pointing, at the next IF statement, at a subprogram entry, or when a specified condition becomes true. The order of execution may be changed simply by pointing with the cursor to a new place in the program and restarting, or by requesting skipping of the next statement or of all statements to the end of the current PERFORM range. New COBOL statements may be added to the program,

BRIEF™

"BRIEF Is The Best Editor I Have Ever Used. I Switched. I Use BRIEF Regularly."

Steve McMahon*

Choosing the right program editor is an extremely important decision. After all, more time is spent with your program editor than with any other category of software. A good program editor helps to stimulate creativity and produce programs of superior design. A poor program editor, on the other hand, can be worse than none at all. It can slow you down, get in your way, and make even the simplest tasks difficult.

BRIEF - PROGRAM EDITING *YOUR WAY*

Every programmer has an individual style that makes their work unique. Most program editors compromise that style by forcing the programmer to conform to their methods. Not so with BRIEF.

BRIEF's most powerful feature is its ability to conform to your way of programming. BRIEF can be easily tailored/customized to your individual preferences. For example, keyboard reassignment allows the keyboard to be used in whatever way you prefer. Keystroke macros allow long sequences of editor directives to be repeated by pressing a single key. Plus you can simultaneously work with numerous program and data files, and configure many windows to control BRIEF's visual presentation.

75% OF THE EXPERTS WHO HAVE TRIED BRIEF SWITCHED!

(Call For Details)

"A BONA FIDE UNDO" Steve McMahon - BYTE REVIEW "BUILD YOUR DREAM EDITOR" MARCH 1985

Here are Steve McMahon's exact words: "...BRIEF implements a true undo facility, by default allowing command-by-command recovery from the last 30 commands...The number of commands you want to be able to undo can be changed." (up to 300) "Only with BRIEF, though, was it possible to undo a macro that produced 4000 words of text with a single keystroke."

EVERY FEATURE YOU WANT

BRIEF supports practically every feature you've ever seen, and some you probably haven't thought of yet.

- Edit Multiple Large Files
- Windows (Tiled & Pop-up)
- Unlimited File Size
- Full UNDO (N Times)
- True Automatic Indent for C
- Exit to DOS Inside BRIEF
- Compile programs inside BRIEF
- Uses All Available Memory
- Intuitive Commands
- Tutorial
- Repeat Keystroke Sequences
- 15 Minute Learning
- Reconfigurable Keyboard
- Online Help
- Search for "Regular Expressions"
- Mnemonic Key Assignments
- Horizontal Scrolling
- Comprehensive Error Recovery
- And ... a Complete, Compiled, Programmable and Readable Macro Language.

```
================================ make.c ================================
int       handle = 0;

main (argc, argv

int             argc;
================ fsa.h ================|================ makefile.h ========
#include "..\include\ctype.h"           |/**
                                        | **             makefile.h:
typedef  struct                         | **
{                                       | **         This is the definitions fil
   short     action,                    | **   Hopefully, it won't be unreasonab
             state;                     | **   that have been written.
} Fsa;                                  | **/
#ifdef   FSA_MAIN
Fsa      fsa[][8] = {/* Alphanum     Co |typedef struct cmd_struct
                                        |{
/* State 0. */         0, 2,         10 |   char         *cmd_text;
/* State 1. */         10, 0,        10 |
/* State 2. */         0, 2,         1, |   struct cmd_struct *next_cmd;
/* State 3. */         0, 5,         11 |} *Cmd_Ptr, Cmd;
/* State 4. */         0, 4,         0, |
Mismatched open parenthesis.          Line: 11    Col: 17        2:17 pm
```

A TYPICAL BRIEF SCREEN

Notice there are three windows on the screen simultaneously and each one is viewing a different file. The mainline of a C program is visible in the uppermost window; the programmer has run a syntax checking macro which found a mismatched open parenthesis in the arguments to the mainline. The other two windows show header files containing information crucial to the design of the program. BRIEF can have an unlimited number of windows and files accessed simultaneously.

BRIEF'S ONLY LIMITATIONS ARE THE ONES *YOU* SET

Full refund if not satisfied in 30 days. ONLY $195

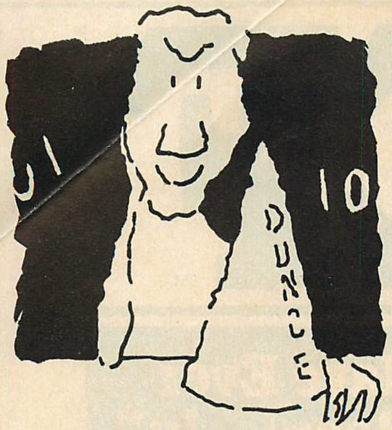
→ **FREE** with order: "Best of BRIEF Macros" — Contest entries include macros for Fortran, C, Calculator, Base Converter, etc. Call before November 30.

AVAILABILITY: IBM-PC and Compatibles, AT, & TANDY 2000

Solution Systems™

335-P Washington St., Norwell, MA 02061 (617) 659-1571

*Steve McMahon's quote courtesy Suntype Publishing Systems. BYTE review by Mr. McMahon may be found in BYTE Magazine March 1985.



but they do not become a permanent part of the source file. A switch controls whether PERFORM and CALL statements are executed as single statements, or whether the perform range or subprogram are stepped into.

The Animator's most useful and entertaining feature, however, is animation, the execution of the program at a reduced speed. As the program executes, the source listing scrolls by on the screen, and each executed statement is highlighted. The speed may be varied through nine levels, and execution may be halted by pressing Ctrl-Break. As Micro Focus puts it, the program "comes to life" on the screen.

During debugging, unformatted (scrolling) screen I/O is displayed on two lines at the bottom of the screen, but full-screen displays are not shown until requested, at which time the screen generated by the program replaces the debugging display. Data values may be displayed by entering the data name or simply by pointing to it with the cursor either in the Data Division or Procedure Division. New data values may then be entered. Any portion of the source program may be displayed, as in an editor, without losing the location of the next statement to be executed. The screen may be divided into two windows to show any two parts of the source file; for example, the Data Division in one and the executing code in another. A search command permits looking for text strings in the source, while another command will locate the Data Division entry of a data or file name pointed to in the program editor, with the cursor positioned where it was last within the debugger.

The Library Module. This module is used to combine multiple source or code files into one; its function is somewhat similar to that of an object module librarian. The combination of many small files into larger ones saves disk space (each individual file may have some wasted space at the end) and improves efficiency because it allows several files to be read after only one directory search and open operation.

A source file library contains copy files that are to be included in source programs during compilation. To bring in a source file from a library, the programmer names the library and the individual files in the COPY statement according to normal COBOL syntax as defined for the Level 2 implementation of the ANSI Library module.

A code file library may be built from any combination of files containing intermediate code, native code, or linked assembly language routines. For COBOL code, the files may contain complete programs or segmented overlays. The use of code libraries in the link step is described below.

Regrettably, there is no provision for maintenance of library files: to remove or replace a member, the programmer must reconstruct the library from its constituent files.

The Build Module. This module performs the functions of a linker. It combines code files and code libraries with the runtime system to build stand-alone COBOL applications that may be run directly from DOS, outside of the Professional COBOL environment. Input to the Build step may be any number of files containing intermediate or native code, or libraries of such files. For library files, only the library directory is included in the output module, and the library members, which may be dynamically loaded subprograms or segmented overlays, are read from the library as needed at runtime. (In contrast, the DOS/Microsoft linker loads subprograms from libraries at link time). When subprograms and segments residing in individual files are linked, they become resident when the application is loaded. Thus the overlay structure may be fine-tuned at link time by including or excluding files.

The Build utility also is provided with Level II COBOL, but there its use is optional because the runtime system is available in an individual file. It is therefore more space-efficient to provide one copy of the runtime for several applications. In Professional COBOL, the runtime system is part of a library file and is not executable outside of the COBOL environment, so each application that runs without benefit of that environment needs to be linked with its own copy of the runtime system.

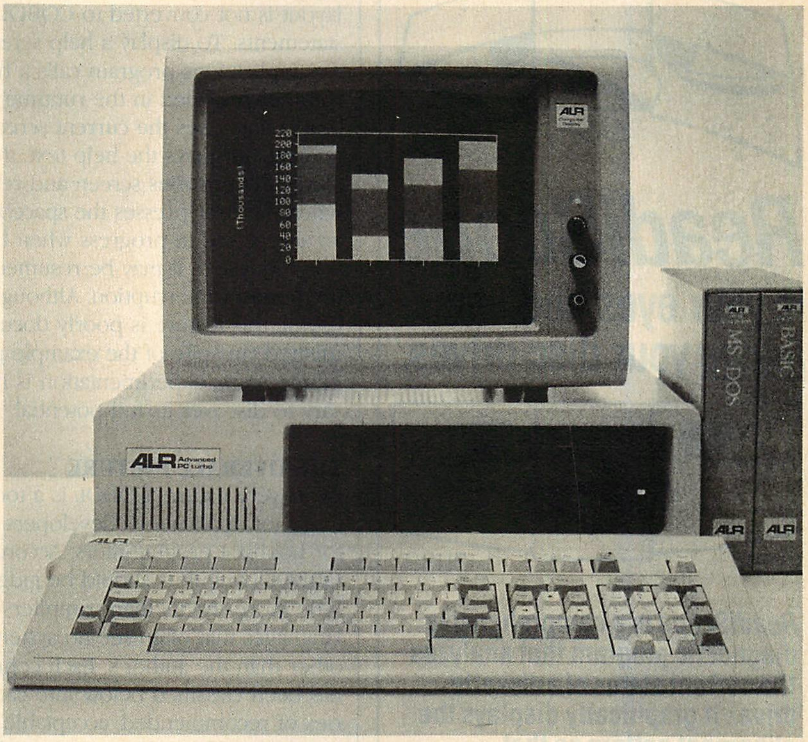
The output of Build is a .COM file that is executable under DOS. If the file size exceeds 64KB, the user may opt to convert it to .EXE form. The conversion does not happen automatically because instead of converting, it might be more efficient to segment the program or exclude subprograms so that the root segment fits within 64KB.

Before leaving Professional COBOL, two other good features deserve mention. The first is a subroutine for defining a set of keys that can terminate full-screen accepts. The keys are specified in a table of key codes; non-ASCII keys such as the function or cursor control keys are identified by extended codes as defined in the *IBM PC*

DEFINING THE IDEAL PC

com • pat • i • ble/kəm-pat-'ə-bəl/1: capable of forming a homogeneous mixture that neither separates nor is altered by electronic interaction 2: capable of operating thousands of programs utilizing the technology of the 8088-2 microprocessor 3: having the ability to run IBM PC and IBM PC/XT programs 4: having the option of exchanging data with the 1.2 Mb IBM PC AT floppy

com • pet • i • tive/kəm-pet-'ət-iv/ 1: inclined, able, or suited to compete 2: competing at **60% faster** with **8 MHz/4.7 MHz**



com • plete/kəm-plet'/1: having all necessary parts, elements or steps 2: leaving little need for add-ons 3: having **20 megabytes** storage and **640k** of memory 4: including Clock calendar, printer and serial ports 5: an enchanced keyboard with separate numeric and curser controls as standard equipment.

con • fi • dence/kan-'fəd-ən(t)s/1: the quality or state of being certain 2: the assurance of a THIRTY-DAY MONEY BACK GUARANTEE 3: having a full ONE YEAR WARRANTY on an entire system

con • clu • sion/kan-'klu-zhan/1: to reach as a logically necessary end by reasoning 2: ideal PC, ALR Advance PC turbo at $1,995 a compatible, competitive, complete system plus the confidence of the ALR guarantee.

con • tact/kan'/takt/1: an establishing of communication with ADVANCED LOGIC RESEARCH, INC. by dialing **(714) 832-7808** on any standard telephone.

ALR

Advanced Logic Research, Inc.

15455 Red Hill Ave., Suite B, Tustin, CA 92680

(714) 832-7808

CIRCLE NO. 104 ON READER SERVICE CARD

Technical Reference manual. After each accept, the program is given the table position of the key that terminated the input. This feature makes it possible to design applications with key-driven command structures.

The other feature allows the implementation of user help screens in COBOL programs. The help screen is designed with the Forms editor, but the layout is not converted to COBOL Data statements. To display a help screen, the applications program calls a help routine (provided in the runtime library) that saves the current screen contents, displays the help text, then restores the previous screen and returns when the user presses the spacebar. If an accept was in progress when help was requested, it may be resumed at the point of interruption. Although very useful, this feature is poorly documented (in spite of the example program); some experimentation is necessary to discover its full potential.

THE OVERALL PICTURE

By its very nature, COBOL is a tool more for applications developers than for hobbyist programmers. Accordingly, COBOL compilers should be judged a little more strictly than compilers for languages that often see no system other than the author's. Each compiler has been classified below into categories of recommended, acceptable, or not acceptable, reflecting its suitability for producing commercial applications. Five of the eight compilers were judged capable of meeting these needs, but some are more capable than others. Because development shops often acquire many copies of a compiler, the price/performance ratio is also important.

In additional, it seems inevitable that a product may fare differently when compared to the entire field than when compared to just one or two others. In the final analysis, the most important judgment is made by those who must live with the product on a daily basis. With these criteria in mind, the following is a best-to-worst ranking.

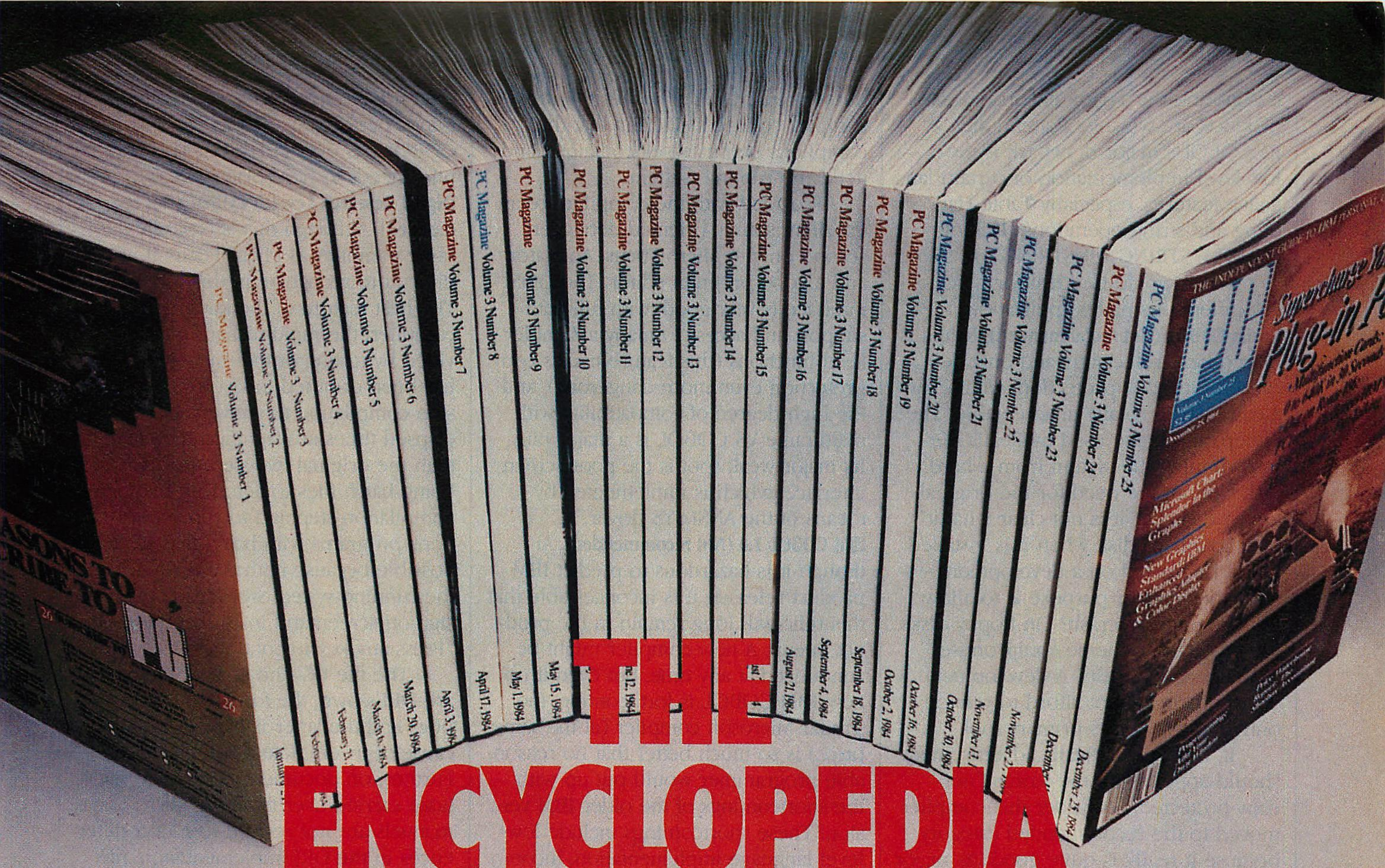
Micro Focus Professional COBOL (Recommended). This product demonstrates the adage that good things do not come cheap. Besides an excellent compiler with a full-featured language implementation, Professional COBOL provides an integrated environment of all the utilities necessary to serious development work. Vendor support sets an example for the industry, the debugger is a marvel, and the whole system is intelligently designed. Except for a few minor inconveniences (primarily the lack of path support) and unless budgetary considerations dictate otherwise, this is the compiler to get.

Micro Focus Level II COBOL (Recommended). The compiler that is the centerpiece of Professional COBOL is good enough to be considered on its own merits. It boasts a quick test turnaround and excellent execution speed; it is disadvantaged by a lack of path support, debugging, and screen layout aids. (Note that Level II COBOL with the Animator debugger and the forms editor is $2,750; if these extras are desired, it is probably worth the additional $250 to get Professional COBOL with its editor, library, and, most importantly, integration.) However, if the utilities are considered superfluous or may be replaced by others, then Level II is a good choice.

Microsoft COBOL 2.0 (Recommended). This compiler just barely squeaks into the recommended group, largely because it is the best of the lower-priced COBOLs. Although $700 is not cheap, it is the lowest price for a full-fledged implementation of all the essential high-level COBOL features. Its slow execution speed in internal operations and poor console I/O interface are compensated by fast file I/O and sorting, and its documentation is the best of that reviewed here. On balance, Microsoft COBOL provides the best performance for the price, and it is especially recommended for users new to the language.

mbp COBOL 9.0 (Acceptable). mbp just misses inclusion in the recommended category. It is usable only on hard-disk systems, and its size and slow compilation speed make it unwieldy even there. This compiler provides respectable execution speed in internal operations, file I/O, and especially the sort, but the latter is not implemented according to the language standard. Other good points are a decent console interface with the forms editor, a usable debugger, and the implementation of program chaining. Using this compiler requires a lot of patience, however; error reporting is especially trying. Apart from price, the relative ranking of mbp and Microsoft COBOLs is largely a matter of subjective judgment about execution speed versus ease of use.

Ryan-McFarland COBOL (Acceptable). There is nothing wrong with RMCOBOL that couldn't be helped by a price cut. At just under $1,000, it simply does not provide the features, either in language implementation or performance, of equally or lower-priced compilers. On merit alone, it is a capable compiler: its small size, convenience of use, and, particularly, its large base of installed appli-



THE ENCYCLOPEDIA OF INFORMATION FOR YOUR IBM PC

Taken one issue at a time, *PC* is the best source of up-to-date news and information about the IBM PC you can find anywhere.

But collect a few, and suddenly you realize that you've assembled a comprehensive reference work covering every significant new IBM PC application, product and development. An encyclopedia that is updated every two weeks.

Each volume of *PC* contributes the latest information on a wide variety of subjects including:

- **New Products** What's the latest in printers, word processing software and database managers? And what's best for your needs? Everything you need to know is in *PC*'s special reports and product reviews.
- **New Applications** Need to design a graphics presentation? Communicate with other PCs? Access a database? Find out how in *PC*'s expertly written articles.
- **New Developments** How are successful professionals and companies using their PCs? Find out in *PC*'s "Pro Columns" and use their proven techniques to move ahead in your own career.

Subscribe to PC today. Build your own IBM PC reference library volume by volume at these special savings:

24 bi-weekly issues for only $21.97.
Less than 92¢ an issue!

To order, complete attached order card or call, toll-free,

1-800-852-5200

PC MAGAZINE

8Z383

cations and third-party enhancement products are worthy of attention. RMCOBOL's major drawback is the low-level implementation of certain language features, especially Table Handling and string operations (lack of SEARCH, STRING, UNSTRING). At a lower price, it would be ideal for quick development of smaller applications.

Realia COBOL (Not recommended). Realia is placed here reluctantly and only because of its copy protection, which, for a compiler, is totally unacceptable. The SuperLok scheme used by Realia permits running the system from a hard disk without the need for the original floppy disk, but it is not clear what it does to a hard disk when it is installed. Can it be trusted on a development system whose major purpose is to allow test programs to crash? On floppy disks, the protection scheme compromises usability and security. A development environment cannot afford to rely on only one copy of its primary tool.

If, however, a Realia advertisement should appear with the words "not copy-protected," consider this compiler moved to the recommended category, because it really is quite good. Its major advantage is its awesome compilation and execution speed: it simply blows away all other COBOLs in everything but relative I/O. It generates the most compact code, considering that the compiler's output is self-contained and needs no runtime system.

Provided with the compiler are an excellent source-oriented debugger, second only to Micro Focus' Animator, and a good program editor. The DOS interface is very good, with full path support and access to many assembler-level functions (this feature makes copy protection even more dangerous), and the high degree of compatibility with mainframe VS COBOL is a major plus. Its major weak spot is the poor screen interface, which is implemented by means of the ANSI.SYS driver.

IBM COBOL 1.0 (Not recommended). Although it is hazardous to predict IBM product releases, it is inconceivable that this relic will long remain in the product lineup. A new compiler might be announced at any time, but even if no new IBM COBOL is forthcoming, the similar Microsoft compiler (at the same price) is so much better that no reasonable programmer would put up with the shortcomings of the older IBM version. These shortcomings include low-level language implementation, incompatibility with large memory, no debugger, erratic compilation of large programs, and an unreliable cursor during screen I/O. These faults outweigh the minor performance advantage that IBM COBOL has over Microsoft. This compiler needs to be replaced.

Digital Research Level II COBOL (Not recommended). Although this is basically the same compiler as Micro Focus II, it is difficult to recognize in this incarnation. The best feature of the Micro Focus compiler, the rapid turnaround provided by testing of intermediate code, has been dropped in favor of a single-step compilation to native code. The elapsed times are significantly longer than the original two-step process. Using batch files to automate the long compilation step and to run applications programs with batch files is not possible because both the compiler and the runtime system stop dead at the sign-on screen and insist that the user "Press any key to continue."

The file I/O and screen interfaces are DRI's own; the former is full of bugs and the latter is barely usable because of incomplete documentation. Further, on documentation, some sections carried over from the Micro Focus manuals are not revised to reflect differences in the DRI implementation: file structures are described incorrectly, nonexistent utilities are mentioned, and sections are missing. This product

All of a sudden, Lotus™ 1-2-3™ is 30 times faster,* safer, and easier.

Recalc+™ for 8087 Now you can accelerate your Lotus 1-2-3™ calculations by an order of magnitude. The new Recalc+ software reroutes your data stream to Intel's 8087 coprocessor, using its dedicated high-speed math capability to dramatically reduce computation time. The more complicated the calculation, the more time you save.

Full-minute recalculations can now run in as little as two seconds (30 times faster).

Best of all, this quantum leap in productivity is just $95 for Recalc+ alone—if you have an 8087 chip. Or $250 for Recalc+ *with* the 8087.

Password+™ Lock your Lotus! The new Password+ electronically locks and unlocks any of your 1-2-3 and Symphony™ worksheet files.

The unique Password+ protection system allows you to enter a password of up to any 36 characters.

Keep your confidential data confidential. For just $49.

Macro+™ It's an ingeniously quicker way to program Lotus. New Macro+ "Learn Mode" displays and stores key strokes as you type them. Now you can create macros in an instant. Macro+ also lets you create and save 10 "hyperspace" macros that do not take up space on your worksheet.

What could be easier than 1-2-3? Macro+. For just $49.

Order toll free today.

All these new programs are available now are not copy-protected. For immediate delivery, just call us and use your credit card.

Call toll-free 1-800-TEAM-GNP.
(From California call 1-800-MEET-GNP)
or contact your computer or software dealer.

GNP

GNP Development Corporation
1244 East Colorado Blvd., Pasadena, Ca. 91106 (818) 577-2550

**Using Intel's 8087 coprocessor, Recalc+ will run calculations up to 30 times faster. Use of Recalc+ and Macro+ requires the use of 1-2-3™ made by Lotus Development Corp.; use of Password+ requires use of 1-2-3 or Symphony™ made by Lotus Development Corp. GNP also offers CPA+, the accounting program for Lotus 1-2-3 and "Team GNP," the quick-help support line for Lotus users. GNP Development Corp. is not associated with Lotus Development Corp. Lotus and 1-2-3 are trademarks of Lotus Development Corp. The 8087 coprocessor is a product of Intel.*

CIRCLE NO. 135 ON READER SERVICE CARD

needs a lot of work, and those hoping to find a full-featured COBOL at a bargain price are advised to consider the other products.

DRI Level II COBOL 3.0: $700
Digital Research, Inc.
P.O. Box 579
Pacific Grove, CA 93950
408/649-5500
CIRCLE 362 ON READER SERVICE CARD

IBM COBOL 1.0: $700
IBM
Entry Systems Division
P.O. Box 1328
Boca Raton, FL 33433
305/982-4700
CIRCLE 363 ON READER SERVICE CARD

mbp COBOL 9.0: $1,000
mbp Software and Systems Technology
7700 Edgewater Drive
Suite 360
Oakland, CA 94621
415/632-1555
CIRCLE 364 ON READER SERVICE CARD

Micro Focus Level II COBOL 2.62: $1,500
Micro Focus Professional COBOL: $3,000
Micro Focus, Inc.
2465 E. Bay Shore Road
Suite 400
Palo Alto, CA 94303
415/856-4161
CIRCLE 465 ON READER SERVICE CARD

Microsoft COBOL 2.0: $700
Microsoft Corporation
10700 Northup Way
Bellevue, WA 98009
206/828-8080
CIRCLE 466 ON READER SERVICE CARD

Realia COBOL 1.2: $995
Realia, Inc.
10 South Riverside Plaza
Chicago, IL 60606
312/346-0642
CIRCLE 467 ON READER SERVICE CARD

RMCOBOL 2.0B: $950; runtime system only, $250
Ryan-McFarland
609 Deep Valley Drive
Rolling Hills Estates, CA 90274
213/541-4828
CIRCLE 468 ON READER SERVICE CARD

Ted Mirecki has a master's degree in computer science and 20 years of experience in information processing. As a corporate planner, he is responsible for developing decision support systems on a variety of hardware.

His company sent him on a mission to make two offices 1800 miles apart, work like one.

Every company has a PC expert like him—responsible for directing microcomputing strategies and budgets. His product recommendations are as good as purchase orders.

Where his mission begins

When he's on a tough assignment, he turns to PC Tech Journal first. Because it gives him the advanced technical information he needs, written by micro experts he can trust.

Inside two recent issues, he finds clues to his company's communication problem: Augie Hansen's review of the new communications package, "Relay"—and a preview of David Schwaderer's long awaited book, "Digital Communications Programming for the IBM PC."

Help him face the challenge

It's not only articles he reads for clues. He explores the information he finds in ads, too.

Ads in PC Tech Journal act as signposts to indicate products that can help him accomplish his task (whether it be a problem involving communications, custom boards, mass storage, or operating systems).

And PC experts find your ad without a lot of digging around. Because every product advertised is listed under its proper category in the product index—just one more reason why sophisticated

C A S E 1 C O M M U N I C A T I O N S

micro experts turn to PC Tech Journal when they're ready to buy.

Accomplish your mission

Don't be the missing clue that can solve this PC expert's problem—and 75,000 other advanced micro experts like him who read PC Tech Journal.

Call your PC Tech Journal sales representative, right now, and let Tech Journal guide customers to where you want them to go—right to your products.

PC TECH JOURNAL

Unmatched power in your market

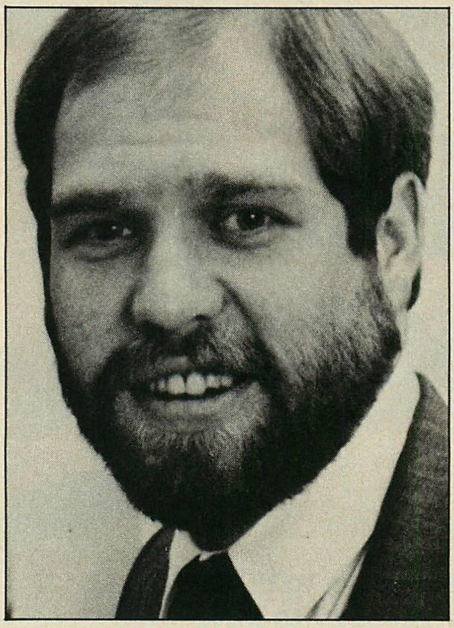
Ziff-Davis Publishing, One Park Avenue, New York, NY 10016 (212) 503-5100

CIRCLE NO. 212 ON READER SERVICE CARD

"I Program In BetterBASIC And I Recommend It."

"I use BetterBASIC over C and PASCAL because it's more versatile and convenient. My department acts as an in-house consultant at Fannie Mae. We evaluate software for the entire organization. When the need arises for a programming language, I recommend BetterBASIC."

Lee Beckley · Information Center Analyst · Fannie Mae · Washington, D.C.



640K Now you can use the full memory of your PC to develop large programs.
STRUCTURED Create well organized programs using procedures and functions that are easily identified and understood and completely reusable in future programs.
MODULAR Use procedures and functions grouped together to form "library modules" which are then available to you or anyone else for future use.
EXTENSIBLE Create your own BetterBASIC modules which contain BetterBASIC extensions. This feature coupled with the easy-to-use Assembly Language support, makes this an ideal OEM language.
INTERACTIVE BetterBASIC acts like an interpreter because it responds to the users' commands in an immediate mode. However each statement is actually compiled as it is entered.
COMPILED Each line of the program is compiled as it is entered into the computer's memory rather than interpreted at runtime.
RUNTIME SYSTEM The optional Runtime System generates stand alone EXE. files allowing for the distribution of products written in BetterBASIC with no royalties.
SUPPORTS Windows, Graphics, DOS and BIOS ROM calls, Chaining, Overlays, Local and Global Variables, Recursion ... and more.
BetterBASIC Runs on IBM PC, XT, AT and all IBM-compatibles. Ask your local dealer for BetterBASIC or call 1-800-225-5800 in Canada call 416-469-5244. Also available for the Tandy 1000, 1200, AND 2000 AT Tandy/Radio Shack stores.

Summit Software Technology, Inc.™
P.O. Box 99, Babson Park
Wellesley, MA 02157

PRICES:

BetterBASIC	$199
8087/80287 Math Module	$99
BTrieve™ Interface	$99
Runtime System	$250
Sample Disk with Tutorial	$10

Better BASIC.

Because It's The Best.

CIRCLE NO. 195 ON READER SERVICE CARD

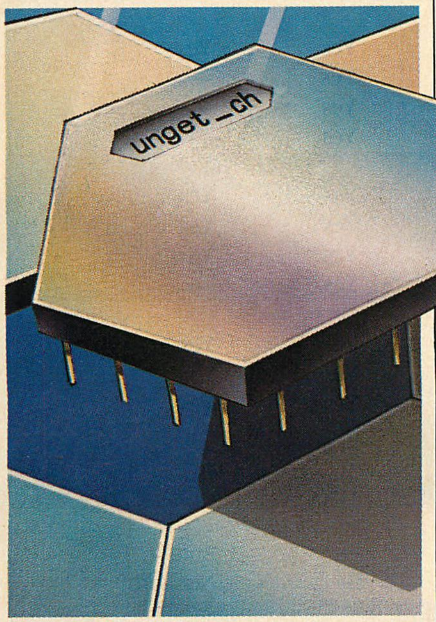
MasterCard, Visa, Checks, Money Order, C.O.D. accepted and P.O. on approval.

(If you're using BetterBASIC and would like to be featured in one of our ads, please write to the Director of Advertising at Summit.)

Only one of these three communications libraries provides enough useful, bug-free functions to be helpful to programmers.

Drop-in Modules for C

WILLIAM J. HUNT



Part 3

Over the past few years, the IBM PC has provided a widely recognized standard for communications hardware in the 8086 arena. Sadly, neither the PC ROM BIOS nor PC-DOS provides adequate software support for this data communications hardware. Implementing good communications software requires extensive knowledge about data communications and the IBM PC hardware, along with good software tools. This article reviews three software libraries for the C language that provide tools for data-communications programming.

In parts 1 and 2 of "Drop-in Modules for C" (William J. Hunt, June 1985, page 100; and William J. Hunt, July 1985, page 135), other libraries from Blaise Computing, Greenleaf Software, and Software Horizons were reviewed. A strong family resemblance exists between those libraries and the communications libraries reviewed here. The Blaise Async Manager library contains a moderate number of useful functions with plenty of documentation. The Greenleaf Communications Library has many functions but with rather disorganized documentation. The Software Horizons Power Pack 3 library also has a large number of functions, but it offers only sparse documentation.

The communications support provided by the PC's BIOS software and by PC-DOS is rudimentary. It includes functions for sending and receiving single characters and for checking correct status. Because these functions provide no buffering, an application must check for and remove each character from the COMM: port as it is received to prevent loss of data. This requirement limits both the baud rate and what an applications program can do. Using an interrupt handler to collect each received character and place it in a buffer or queue frees the applications program and allows higher-speed communications. Providing interrupt-driven, buffered character output is not as important, but it can improve the effective speed of character output.

Most communications applications need to make connections over telephone lines before data can be transferred. The modems sold by the D. C. Hayes Company have established a de facto standard for dialing and answering calls. These modems and compatible models sold by other companies are in use on many PCs and are supported by many communications programs for the PC. Library functions to control a Hayes-compatible modem reduce the effort needed to implement telephone line support in an application.

Telephone lines were not designed to transfer data rapidly and without errors. An additional protocol for detecting errors and correcting them is a necessity in many applications. Although the basic PC hardware supports generating and checking a parity bit, these capabilities are inadequate for many applications. Protocols must be supported at both ends of a connection to be useful. (For example, the XMODEM protocol, which was originally defined by Ward Christiansen in the late 1970s, is supported by a number of communications programs on the PC and by data-retrieval services such as CompuServe.) Such protocols require specialized knowledge and programming skills to

ILLUSTRATION • ANDY ZITO

implement well; a good implementation of the XMODEM protocol is a valuable part of a communications library.

The products reviewed support asynchronous communications using the IBM PC async communications adapter or equivalent hardware. The libraries all provide interrupt-driven data reception, RS-232 modem control and status monitoring, control for a Hayes modem, and file transmission and reception with the XMODEM protocol.

Table 1 lists some general information about the communications libraries reviewed. Microsoft C refers to versions 2.03 and earlier, which are based on Lattice C (the compiler used in this review). At the time the review was completed, none of the products supported version 3.0 of Microsoft C. The table also lists a number of features a good communications library might include. A yes entry means that the product provides the feature in a generally useful way. None of the three libraries requires a runtime fee.

Interrupt-driven data reception is a necessity for useful applications; each library provides that feature. The Blaise and Greenleaf libraries also provide interrupt-driven data transmission. Noninterrupt-driven transmission and reception may be useful sometimes; each product provides these functions. Sometimes it is convenient to send or receive a character string or an array of characters with one function call. Such a function also may produce better efficiency.

All of the libraries provide functions to collect modem and error status and to check for characters waiting in the received data queue. Only the Greenleaf library provides a function to check for the successful transmission of characters queued for output.

Minicomputers and mainframes are often unable to accept a stream of input at the full baud rate; software in a personal computer must pace output when communicating with a larger computer. No library provided this pacing feature. The option to rely on or ignore modem status input is essential; modems, computers, and other devices using the RS-232 interface vary widely in the way they use modem signals.

Choosing appropriate sizes for queues for received data and output data often is necessary to achieve a good performance in a communications application. The Software Horizons library uses a fixed 1KB input buffer size, but the other libraries allow the programmer to specify buffer sizes.

The Blaise library supports Xon-Xoff flow control automatically in both

TABLE 1: *Product Features*

	ASYNC MANAGER	GREENLEAF COMM LIBRARY	POWER PACK 3
COMPILERS SUPPORTED			
Lattice	Yes	Yes	Yes
Microsoft	Yes	Yes	Yes
Computer Innovations	Yes	Yes	Yes
Mark Williams	—	Yes	Yes
DeSmet	—	Yes	Yes
Digital Research	—	—	Yes
SOURCE CODE INCLUDED	Yes	Yes	Yes
SUPPORT FOR ALL MEMORY MODELS	Yes	Yes	Yes
FEATURES INCLUDED			
Initializes COMn:	Yes	Yes	Yes
Supports COM1: and COM2:	Yes	Yes	Yes[a]
Supports more than one port at once	—	Yes	—
DATA TRANSFER			
Noninterrupt			
Rcv char	Yes	Yes	Yes
Xmt char	Yes	Yes	Yes
Interrupt-driven			
Rcv char	Yes	Yes	Yes
Rcv string/array	Yes	Yes	—
Xmt char	Yes	Yes	—
Xmt string/array	Yes	Yes	—
Status			
Modem status	Yes	Yes	Yes
Error status	Yes	Yes	Yes
Rcv queue empty	Yes	Yes	Yes
Xmt completed	—	Yes	—
Control			
Modem control	Yes	Yes	Yes
Clear Rcv queue	Yes	Yes	Yes
Clear Xmt queue	Yes	Yes	Yes
Options/flexibility			
Output pacing	—	—	—
Ignore modem status	Yes	Yes	—
Flow control (Xon/Xoff)			
Local	Auto	—	Auto
Remote	Auto	—	Not auto
HAYES MODEM CONTROL			
Format a command	Yes	—	Yes
Get a response	Yes	Yes	Yes
Specific commands (dial, answer)	—	Yes	Yes
Send attention string ("+++")	—	Yes	Yes
TERMINAL EMULATOR SKELETON	—	—	Yes
XMODEM SUPPORT			
Send file	Yes	—	Yes[a]
Rcv file	Yes	—	Yes[a]
Send block	Yes	Yes	—
Rcv block	Yes	Yes	—
Set timeout and retry count	Yes	Yes	—

[a] The feature is present but does not work properly.

In this table, Microsoft C refers to versions 2.03 and earlier, which are based on Lattice C. At this writing, no product supports version 3.0 of Microsoft C.

directions. Flow control for received data (local flow control) is automatic for the Software Horizons library; remote flow control (responding to flow control messages from the remote end) must be implemented using tools in the library. The Greenleaf library does not provide automatic support for flow control or tools for implementing it.

Many commands are recognized by a Hayes modem; table 1 records only whether the libraries support these commands. The Greenleaf and Software Horizons libraries provide functions for all commands, and the Blaise library provides a single function that places the standard prefix and terminator on a command supplied by the user.

The Software Horizons library is unique in providing the skeleton for a terminal emulator as a function. This library function implements a loop to check for keyboard input and received data, calling functions supplied by the user whenever action is needed.

All three libraries provided some functions to support the XMODEM protocol. The Blaise library functions were more flexible and easier to use than those in the other two libraries.

INFORMATION, PLEASE

Although good documentation is important for any sort of tools library, it is especially important for the communications support libraries reviewed here. No matter how well a communications library is designed and implemented, the library's users will encounter certain practical problems for which the library should provide some guidance.

Table 2 rates libraries on the effectiveness of the documentation they provide. The following criteria were used to establish what constitutes good documentation. The manual should have clear step-by-step installation instructions with a correct inventory of files on the distribution disks and an identification of those files necessary to routine compiling and linking. Operational instructions should describe steps needed for compiling and linking C programs to use library functions.

The manual should explain the basic concepts behind the communications functions. Some topics that should be discussed are listed below:

Sequence of operations. Every library tested requires a series of function calls to prepare for transmitting or receiving data. Those sequences should be illustrated in the manual with an example and a step-by-step explanation.

Using interrupts. The manual should provide some description of how asynchronous port interrupts work on the IBM PC. The need for removing the asynchronous port interrupt handler before exiting a program using library functions should be made clear.

Choosing communications buffer sizes. Guidelines for setting sizes of buffers for transmitting and receiving data should be explained. For example, writing received data to a floppy disk might take a half second. At 1200 baud, 60 additional characters might be received while the PC waits for the disk I/O operation to be completed. At 9600 baud, 480 characters would be received. Thus, to prevent loss of received data, writing data to a floppy disk might require a buffer size of 60 characters for 1200-baud operation and 480 characters for 9600-baud operation.

Xon-Xoff flow control. The concepts of local and remote flow control should be explained. Examples that illustrate how to implement flow control with the library functions should be presented.

Hayes modem operation. The manual should provide short, thoroughly explained examples of using a Hayes

modem for dialing a call, waiting for and answering a call, and sensing and setting modem parameters. Some information on the format of Hayes modem commands and responses and the sequence of operations for using each command is also important. Library functions may require particular settings of some Hayes modem parameters—these requirements should be stated.

XMODEM protocol use. When library functions are used to communicate with another system using the XMODEM protocol, differences in implementation of the protocol between the program that sends the data and the software that receives it may cause problems. The manual should discuss possible differences and show how to modify XMODEM parameters (such as timeouts and limits on number of retries).

The manual should document each individual function with a thorough description of what it does and how to use it. Functions related to or required by the one currently being described should be listed with it.

As the overall ratings show, none of these products provides enough documentation, although the Blaise library is better than the other products.

The authors of these libraries have made an assumption that the user knows quite a lot about data communications and its practical problems. To use any of these libraries in applications programs, the programmer will have to go elsewhere for information on how asynchronous communications work and the practical problems that he will face in using data communications.

Good performance is important in most data communications applications. Because available modems offer rather slow data-transfer rates, the time required to transfer a useful amount of data is often a limiting factor in the performance of the application as a whole. In addition, high costs for telephone-line connect time make efficient data transfer a necessity. Finally, although communications via an RS-232 cable instead of a modem may not have the same costs for connect time, it may require operation at 9600 baud instead of 1200 or 2400 baud. This section presents some programs that measure data-communications performance and gives results for the three libraries.

Because communications applications involve two systems—one at each end of a connection—these benchmark programs come in pairs. To control conditions and produce a precise measurement, the tests use two IBM PCs with a test program in each.

TABLE 2: *Documentation*

	ASYNC MANAGER	GREENLEAF COMM LIBRARY	POWER PACK 3
INSTALLATION	Fair	Fair to good	Fair to good
OPERATION	Fair	Fair	Poor to fair
TOPICS	Fair	Very poor	Poor
INDIVIDUAL FUNCTIONS	Good	Poor	Poor
OVERALL QUALITY	Fair to good	Poor to fair	Poor to fair

None of the products reviewed here provides satisfactory documentation, although that from Blaise is better than that from any of the other companies.

TABLE 3: *Benchmark Results*

	ASYNC MANAGER	GREENLEAF COMM LIBRARY	POWER PACK 3
XMT—RCV TEST			
Max. baud rate	9600	Failed	4800
Secs per 1,000 chars	1.25	test	2.1
XON-XOFF TEST			
Flow control worked	Yes	Not supported	Failed test
HAYES MODEM TEST			
Dial	Yes	Yes	Yes
Answer	Yes	Yes	Yes
Read/set modem parms	Yes	Yes	Yes
Hang up	Yes	Yes	Yes
XMODEM TEST			
(secs per 1,000 chars)			Failed test[a]
at 9600 baud	5.7	4.4	1.2
at 4800 baud	8.9	5.4	2.5
at 2400 baud	8.8	7.3	4.9
at 1200 baud	13.2	12.0	9.8
at 300 baud	39.1	41.5	39

[a] *Power Pack 3 failed the XMODEM test; these figures are possible results only.*

To control conditions and to produce a precise, reproducible measurement, each of these communications benchmarks used two IBM PCs with a test program in each.

Xmt-Rcv Test. The program in listing 1 sends a file out on the COM2: port. It transmits ASCII characters without any special protocol. Listing 2 lists the matching program that receives the file. Together the programs measure how well communications-library functions transmit and receive data. Transfer time is measured with the timer function (renamed **eltime** to avoid interference with functions in the communications libraries). Marking the start of the file with a capital *S* and the end of the file with a capital *Q* permits precise timing in both the transmit and receive programs. The programs in listings 1 and 2 use the Blaise Async Manager library functions. Similar versions were written for use with the other libraries.

The test programs measure the effective speed achievable by the library functions. They also show whether the library functions can handle received characters at a high baud rate.

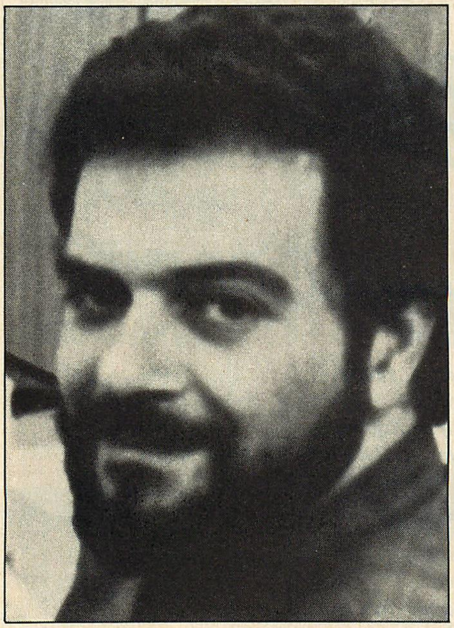
Results of using these programs are shown in table 3. The highest speed at which a 30,001-character file could be transferred successfully is reported, as is the time per 1,000 characters. The Blaise version operated successfully at 9600 baud, and the Software Horizons version worked at 4800 baud but lost characters at 9600 baud. A bug in the Greenleaf library functions caused data to be lost by the program transmitting the file. The bug may have occurred when the output buffer maintained by the library functions was filled.

Xon-Xoff Protocol. When the program in listing 2 is modified to display each character received with the C library function **putchar**, it cannot keep up

"I Program In BetterBASIC And I Recommend It."

"It would have taken me four months in FORTRAN to write DEA's program which is a sophisticated data base and process control system that regulates all the processes used in our circuit board manufacturing etching line. But in BetterBASIC I wrote the program in a much shorter time. I use BetterBASIC and I recommend it."

Claudio Edelman · DEA Equipment · Tempe, Arizona



640K Now you can use the full memory of your PC to develop large programs.
STRUCTURED Create well organized programs using procedures and functions that are easily identified and understood and completely reusable in future programs.
MODULAR Use procedures and functions grouped together to form "library modules" which are then available to you or anyone else for future use.
EXTENSIBLE Create your own BetterBASIC modules which contain BetterBASIC extensions. This feature coupled with the easy-to-use Assembly Language support, makes this an ideal OEM language.
INTERACTIVE BetterBASIC acts like an interpreter because it responds to the users' commands in an immediate mode. However each statement is actually compiled as it is entered.
COMPILED Each line of the program is compiled as it is entered into the computer's memory rather than interpreted at runtime.
RUNTIME SYSTEM The optional Runtime System generates stand alone EXE. files allowing for the distribution of products written in BetterBASIC with no royalties.
SUPPORTS Windows, Graphics, DOS and BIOS ROM calls, Chaining, Overlays, Local and Global Variables, Recursion ... and more.
BetterBASIC Runs on IBM PC, XT, AT and all IBM-compatibles. Ask your local dealer for BetterBASIC or call 1-800-225-5800 in Canada call 416-469-5244. Also available for the Tandy 1000, 1200, AND 2000 AT Tandy/Radio Shack stores.

Summit Software Technology, Inc.™
P.O. Box 99, Babson Park
Wellesley, MA 02157

PRICES:

BetterBASIC	$199
8087/80287 Math Module	$99
BTrieve™ Interface	$99
Runtime System	$250
Sample Disk with Tutorial	$10

Better BASIC.

Because It's The Best.

MasterCard, Visa, Checks, Money Order, C.O.D. accepted and P.O. on approval.

(If you're using BetterBASIC and would like to be featured in one of our ads, please write to the Director of Advertising at Summit.)

CIRCLE NO. 169 ON READER SERVICE CARD

with characters received at 9600 baud or even 2400 baud. This provides an effective test of flow control using the Xon-Xoff protocol. The transmit and receive programs were modified to enable the flow control (these modified programs are not shown). Only the Blaise library provided flow-control support that worked. The Greenleaf library provides little support for flow control and little documentation on implementing it using tools in the library. The Software Horizons library provided support for flow control, but it did not work properly. Table 3 records whether flow control worked properly—the transfer speed is controlled by the slow speed of displaying characters through the PC's ROM BIOS services.

Hayes modem control. Functions for dialing out, answering an incoming call, setting modem parameters, and hanging up (disconnecting) a call were checked. Performance is not an issue, so table 3 reports only whether the library functions operated correctly.

XMODEM protocol. Listings 3 and 4 show programs to transmit and receive a file using the XMODEM protocol. The versions shown are for the Greenleaf Library. The programs contain functions to send and receive a file; versions of the programs in the Blaise and Software Horizons libraries used library functions to send and receive a file.

Table 2 shows times for transferring the same 30,001-byte file at different baud rates. The last column gives the best possible time that might be achieved by an efficient XMODEM implementation. (This time corresponds to 85 percent of the nominal speed—that is, at 9600 baud, it represents 85 percent of 960 characters per second.)

The benchmarks were run on two IBM PCs. To allow for precise, repeatable tests, the PCs were connected by an RS-232 cable rather than by modems and telephone lines. The files transferred were located on RAM disks. Buffer sizes of 1,000 bytes were used for input and output in all versions of the benchmark programs.

READING SOURCE CODE

Designing and executing tests to find bugs in communications software can be time-consuming. Because the testing time here was limited, the testing process was supplemented by a reading of the source code for the functions. Some observations from these reading regarding the source code follow.

None of the libraries implemented the XMODEM protocol well. Neither were any good examples presented,

and the implementations provided here cannot be considered as starting points for implementing other protocols. In addition, because none of the libraries was coded in a modular style, mistakes were repeated in several places, making it difficult to fix problems.

Every library handled timeout intervals poorly in the XMODEM protocol. The Blaise library used the following logic to receive an acknowledgment or the first character of a data block:

```
do {
    check for a received char
    if (present)
    {get the character
    return
    }
    wait for an interval
    retry count = retry count + 1
    } while (retry count < limit ) ;
```

This sort of idle delay should never be used to implement timeout intervals. The logic should alternate between checking the elapsed time and checking for the presence of a character, returning immediately when a character is found or the timeout interval reached.

Another problem concerning timing was exhibited by the Greenleaf library. It used a delay loop based on instruction execution timing in transmitting a break signal. The Software Horizons library used such a loop for setting a timeout interval for received characters in the XMODEM protocol. Such loops, which are based on the timing of instruction execution, depend directly on processor speed. A loop that creates a 200-millisecond interval on an IBM PC might create an interval of 80-100 milliseconds on an IBM PC/AT. Using such loops for timing is an unacceptable programming technique.

Some bugs showed up in the implementations of the XMODEM protocol. When this protocol is used, telephone line errors can cause acknowledgments of data blocks to be lost. The computer sending a file would retransmit the same data block if an acknowledgment was lost. When the computer receiving a file receives the same data block a second time, it must return a positive acknowledgment again. (Of course, it should not store the block in the file again.) The Greenleaf and Software Horizons libraries both made errors in handling this situation. The Greenleaf library functions reject the retransmitted block, sending a negative acknowledgment. The Software Horizons library functions return a positive acknowledgment for any block number if more than 32KB have been received.


TECH JOURNAL
FOUR PASCALS FOR THE IBM PC
Rating compilers from Borland, DRI, Microsoft, and SBB
RESEARCH REPORT: DATA MODELS
COMMUNICATIONS REVIEW: CROSSTALK XVI
MODEMS DEMYSTIFIED
MATHEMATICAL MODELING ON THE IBM PC WITH TUTSIM
STRUCTURED BASIC
PASCAL

PROGRAMMER DEVELOPMENT TOOLS

TURBO PASCAL AND UTILITIES

	List	Ours
Turbo PASCAL *ver 3.0* by Borland Int'l .... *Sale*	70	49
Turbo PASCAL *w/8087 or BCD* .... *Sale*	110	89
Turbo PASCAL *w/8087 & BCD* .... *Sale*	125	99
Btrieve by Softcraft	250	199
ESP for Pascal by Bellesoft	Call	Call
MetaWindows *for Turbo PASCAL* by MetaGraphics .	55	49
Multi-Halo Graphics by Media Cybernetics	250	199
Screen Sculptor by Software Bottling	125	109
Turbo ASYNCH by Blaise Computing	100	89
Turbo GRAPHICS TOOLBOX by Borland Int'l.	55	49
Turbo POWER TOOLS by Blaise Computing . *New*	100	89
Turbo TOOLBOX by Borland Int'l	55	49
Turbo TUTOR by Borland Int'l	35	29
TurboPower Util w/source by TurboPower Sftwr ..	95	89
XTC Text Editor by Wendin	99	89

BASIC LANGUAGE

	List	Ours
BetterBASIC by Summit Software	200	169
8087 Math Support	99	89
Run-time Module	250	239
Professional BASIC by Morgan Computing	99	89
8087 Math Support	50	47
True Basic from Addison-Wesley .... *New*	150	129
Run-time Module	500	459

OTHER LANGUAGES

	List	Ours
8088 Assembler w/Z-80 Translator by 2500 AD ...	100	89
APL*PLUS/PC by STSC	595	449
Golden Common LISP by Gold Hill	495	Call
Janus/ADA by R&R Software	900	699
MASM-86 ver 3.0 *w/utilities* by Microsoft	150	109
Modula-2/86 by Logitech	495	439
MS Fortran 3.3 .... *Links with MS C 3.0*	350	239
MS Pascal 3.3 .... *Links with MS C 3.0*	300	219
Prolog VMS by Automata Design Associates *New*	100	Call
Prolog VML by Automata Design Associates *New*	300	Call
Prolog VMA by Automata Design Associates *New*	500	Call
Prolog-86 by Solution Systems	125	Call
RM/Fortran by Ryan-McFarland	595	439

OTHER LANGUAGE SUPPORT PRODUCTS

	List	Ours
APL2C by Lauer Software ..... *Interfaces APL to C*	150	139
Btrieve By SoftCraft	250	199
Blaise Tools *for Pascal*	Call	Call
FORTRAN Libraries by Alpha Computer Service .	Call	Call
Sci Subroutine Lib *for Fortran or Basic*	175	139

OTHER PRODUCTS

	List	Ours
Advanced Trace-86 by Morgan Computing	175	149
Codesmith-86 Debugger by Visual Age	145	129
Polytron Products .... *We Carry a Full Line*	Call	Call
Profiler by DWB Associates	125	89
Rtrieve by Softcraft	85	79
Xtrieve by Softcraft	195	169

Periscope Symbolic Debugger
by Data Base Decisions

Write-protect memory board and breakout switch allows instant recovery from runaway code. Provides on-line help, windowing, extensive breakpoints, dual monitor support and more.

List Price $295 Our Price **$269**

PHOENIX PRODUCTS

SUMMERTIME SALE!

In stock and ready for immediate shipping.

	List	Ours
Pasm86 *Macro Assembler*	295	195
Pfinish *Performance Analyzer*	395	269
Pfix-86 Plus *Symbolic Debugger for Plink-86*	395	269
Plink-86 *Overlay Linker*	395	269
Pmate *Macro Text Editor*	225	145
Pmaker *Program Development Manager*	195	119
Pre-C *Lint Utility for C*	395	269

Dealer Inquiries Invited

C LANGUAGE

	List	Ours
C-terp *C Interpreter* by Gimpel Software	300	269
C-terp *demo system* .... *Applied to purchase*	45	45
Computer Innovations C-86 Compiler	395	299
DeSmet C Compiler with Debugger	159	145
Instant C by Rational Systems .... *Sale*	500	399
Lattice C Compiler from Lattice	500	349
Lattice C from Lifeboat .... *Ltd Qty Special*	500	275
Mark Williams MWC-86 w/Source Debugger . *Sale*	495	379
Safe C Standalone Interpreter by Catalytix	400	Call
Wizard C Compiler by Wizard Systems ...... *Sale*	450	359
Xenix Development System by SCO	1350	1099

MACINTOSH C COMPILERS

	List	Ours
DeSmet/Ouye C Compiler .... *New*	150	139
Mac C by Consulair .... *Call for Details*	Call	Call
Megamax C *compiler for MacIntosh*	295	239

MICROSOFT C 3.0 AND UTILITIES

	List	Ours
Microsoft C Compiler version 3.0 .... *Sale*	395	259
Blaise C Tools	125	109
Blaise C Tools 2	100	89
C-terp by Gimpel Software	300	269
C Utility Library by Essential Software	185	139
Greenleaf C Functions Library ver 3.0	185	139
Greenleaf Comm Library	185	139
The HAMMER by OES .... *New*	195	179
Multi-Halo Graphics by Media Cybernetics	250	199
PANEL *Screen Designer ver. 6.0* by Roundhill	295	234
C Power Paks from Software Horizons	Call	Call
Windows for C by Creative Solutions	195	139

C UTILITIES

	List	Ours
Basic_C Library by C Source	175	139
Blaise C Tools	125	109
Blaise C Tools 2	100	89
Btrieve by SoftCraft	250	199
C Power Paks From Software Horizons	Call	Call
c-tree by FairCom .... *New version*	395	359
C Utility Library by Essential Software	185	139
ESP for C by Bellesoft	Call	Call
GraphiC by Scientific Endeavors ..... *New version*	250	209
Greenleaf C Functions Library ver 3.0	185	139
Greenleaf Comm Library	185	139
Multi-Halo Graphics by Media Cybernetics	250	199
PANEL *Screen Designer ver. 6.0* by Roundhill	295	234
PC Lint by Gimpel Software	100	89
Safe C Dynamic Profiler by Catalytix	150	Call
Safe C Runtime Analyzer by Catalytix	400	Call
Scientific Subroutine Lib *for C* by Peerless	175	139
Windows For C by Creative Solutions	195	139

The **HAMMER** C Library by OES

This excellent new C library is designed for creating end-user interfaces. There are functions for creating 123-like menus, managing the screen, creating input forms, prompting for inputs and more. No royalties and includes source code.

List Price $195 Our Price **$179**

LATTICE PRODUCTS

All products receive support and updates directly from Lattice. Lattice C is in stock and ready for shipment.

	List	Ours
Lattice C Compiler from Lattice	500	349
C-Food Smorgasbord	150	119
C-Sprite *Program Debugger*	175	149
Curses *Screen Manager*	125	109
dBC *C Interface for dBase II or III Files*	250	209
dBC *with source code*	500	439
LMK *Make Facility*	195	159
Text Mgmt Utils *(GREP/DIFF/ED/WC/Extract/Build)* ...	120	105

TEXT EDITORS

	List	Ours
Brief By Solution Systems	195	Call
Epsilon *Emacs-like Text Editor* by Lugaru	195	179
ESP for C *or* **Pascal** by Bellesoft	Call	Call
ESP for C *and* **Pascal** by Bellesoft	Call	Call
Vedit by Compuview	Call	Call
XTC Text Editor By Wendin	99	89

Programmer's Connection

136 Sunnyside Street Hartville, Ohio 44632 (216) 877-3781 (In Ohio)

U.S.; **1-800-336-1166** Canada; **1-800-225-1166**

Call For Our Catalog

Account is charged when order is shipped
Prices are subject to change without notice

BLAISE ASYNC MANAGER

The Async Manager from Blaise Computing shares some of the virtues of the company's Ctools and Ctools2 products. And, although its documentation is not as good as theirs, it does exhibit some of the same strengths.

Individual functions are well documented, with short usage examples at the end of each description. The table of contents lists functions with a one-line description to help users find the functions they need. Functions are grouped by the level of service they provide, and each of these sections begins with a discussion of topics common to that group of functions.

The documentation is less successful in other areas, however. Separate installation and usage instructions are provided for each level of functions. A single, integrated discussion would be much more useful. Although the topical discussions provided in the manual are well done and useful, a number of important topics are not discussed. The manual contains no explanation of the sequence of operations needed to set up and use the PC's communications hardware. Some usage examples for individual functions provide partial information about the sequence, but a full explanation is really necessary.

Like the other libraries reviewed, the Blaise distribution disks include several sample programs. These programs illustrate the use of library functions, but they are too long and complicated to be effective as examples. Shorter, simpler example programs with full explanations should have been included in the manual.

The source code for the Blaise library is well commented. A utility program provided on the distribution disk prints the source files in the order that functions are documented in the manual. This makes it easy to find the source code for a function. The utility puts tab characters in the output; some printers, an old EPSON MX-80 for example, will ignore them and produce a listing with incorrect indentation.

Its performance in transmitting and receiving data is good. Remote and local flow control are handled efficiently and automatically by the library functions. XMODEM file transfer is not very efficient, but it works correctly.

The library provides minimal support for controlling a Hayes modem. One library function sends a command to the modem—the user specifies the command as a character string and the library function sends a standard prefix ("AT), the string, and a terminator (carriage return). Another function waits for a certain amount of time and then returns response characters received from the modem. This function is satisfactory when the Hayes modem returns single-digit responses but awkward when it is in "verbal" (English-language responses) mode. No information is provided about specific Hayes commands; the Hayes manual must be consulted for that information. What the Blaise library functions provide is trivial and misses the most important information about using a Hayes modem.

To its credit, the Blaise Async Manager does a good job of sending and receiving data. Its implementation of the XMODEM protocol is mediocre, but the functions are easy to use. Its support for the Hayes modem has little value. The Blaise documentation is not entirely satisfactory but is certainly the best of the three.

THE GREENLEAF COMM LIBRARY

The Greenleaf Communications library is not a finished product. The library functions contain obvious bugs, and the documentation does not match what the functions really do.

Installation and operation instructions are adequate, but the rest of the documentation is unsatisfactory. Descriptions of most individual functions contain usage examples, but some descriptions simply say "See the demo programs." Possible values for function arguments and values returned by the functions are poorly documented.

The sequence of functions calls needed to set up and use communications ports is not explained in the manual. A documentation file on the distribution disk lists the sequence of calls without explanation. Some usage examples for individual functions contain examples of set-up and initialization operations, but these examples are not explained, either.

Some functions are rendered useless by incomplete or misleading documentation. For example, the description of the **asigets** function (it stores received characters in a string) contains the following two sentences: "Zero or more characters are transferred from the Receiver ring buffer for the port to the specified string." and "No data are received by this function (it just provides the illusion thereof.)"

Descriptions of functions to transmit characters and to collect received characters do not mention the effect of error conditions or of modem status bits on the functions. The **asigetc** function may return a -1 value if an error

flag has been turned on, but the manual does not indicate that. Important questions are not answered. For example, the **asigets** function places received characters into a C string with a null character to mark the end of the string. What happens when a null character is received? The function stores the received null character in the string, places an additional null character after it, and returns immediately with a normal return value. The manual says nothing about this situation.

The library includes 37 functions for controlling a Hayes-compatible modem. Seven pages of documentation describe how the individual functions work but provide no information on Hayes modems or how the functions are used to control them. The manual states that library functions were tested with default switch settings for the modem, and it contains a warning that other settings might produce unpredictable results; unfortunately, the manual does not state which switch settings the library functions require. For example, the switches must be set for single-digit response mode and no local echo, but nothing in the documentation indicates such.

Documentation of functions that support the XMODEM protocol is even more inadequate. The manual does discuss the format of XMODEM messages and gives an example of the sequence of data block and acknowledgments required to transmit a file. However, the descriptions of individual functions give no sense of what they actually do or how individual functions work together.

In fact, much of what the manual says about XMODEM support functions does not match the way those functions actually work. For example, the manual describes a function **asixsendbuf** designed to send a user buffer using XMODEM. This description does not mention that the function also waits for an acknowledgment, retransmitting the data block if necessary. Another function, **asirecbuf**, waits for and collects a data block; it also sends negative acknowledgments until a block is received correctly; again, the manual does not indicate this. A documentation file on the distribution disk does discuss changes to library functions, but that discussion is not an adequate replacement for discussions in the manual.

The manual contains several disclaimers about the XMODEM and Xon/Xoff protocols:

> It is up to the user to provide the fine interpretations of the protocol definitions.
>
> We do not pretend, however, to supply a complete protocol (for ANY protocol, standard or nonstandard). As it turns out, 'Xon/Xoff' sounds simple, but there are hundreds of small variations on the theme which go to prove the notion that trying to support something like this in a product is possible only when you control both ends of the line.

The manual fails to describe what "fine interpretations" or "small variations" the user might encounter. The problems the manual's authors describe are real, but Greenleaf's customers deserve some information on what variations might be encountered and how to cope with them. As it stands, Greenleaf has done the easy part of the job and left the user to finish the hard part very much on his own.

To really understand how the Greenleaf library's functions work, the user must read the source code. The manual helps by listing the source file in which a function may be found. The source code itself is not very modular and not well commented.

Like the other libraries reviewed, the Greenleaf library relies on a long, poorly commented sample program (**term.c**) to document the use of library functions. A few test programs also are included on the disk as examples. Neither the manual nor the documentation files on the distribution disk explain the sample programs.

The Greenleaf library offers some unique features. It supports use of more than one communications port at the same time. It also supports the use of COM3: and COM4: communications ports. Unfortunately, port addresses and interrupt levels for COM3: and COM4: are not standardized, so this capability has limited usefulness.

Some of the functions in the Greenleaf library are poorly designed. For example, the **asiputc** function places a character string into the transmit queue for the transmit interrupt routine to send out. If there is room for the entire string in the transmit queue, the function returns a zero value. If the queue becomes full before all characters in the string have been placed in the queue, the function returns a −1 error value. What it does not indicate is how many characters were placed in the queue. The user has the choice of calling **asiputs** again and transmitting some of the characters twice or not calling **asiputs** again, which guarantees that some characters will not be transmitted at all. An input function called **asigetb** fills a buffer array with characters from the received data queue; if the queue is emptied before the entire buffer is filled, the function returns an error value but does not give the number of characters actually transferred.

The benchmark for transmitting and receiving characters failed as a result of a bug in the library's character transmission functions. Results for the XMODEM transfers are better than for the Blaise library but both are slow.

Overall, the Greenleaf Communications Library cannot be recommended. It is poorly implemented with serious bugs. The manual provides insufficient information to understand and use the library functions—function descriptions are incomplete or misleading.

POWER PACK 3 LIBRARY

The Software Horizons Power Pack 3 library (PP3) is part of a series of tools libraries produced by this company; the Power Pack 1 and 4 libraries were reviewed in parts 1 and 2 of this article. Unlike those libraries, which provided support for several types of functions, PP3 concentrates on communications.

The PP3 library makes use of some functions from the basic PP1 product. Those functions are provided with the PP3 product so that the user need not purchase the PP1 product. However, because the PP3 manual does not document the functions that it uses from PP1, it might be necessary to buy the PP1 library in order to get documentation for those functions.

Like the Power Pack libraries reviewed earlier, PP3 is long on functions and short on documentation. Very few examples are provided in the manual and there is little discussion of the sequence of operations needed to set up and use the PC's communications hardware. XMODEM support is especially complicated to use—the user must supply his own functions to display messages and to accept yes or no inputs from the keyboard. The manual discusses the requirement, but without a short, clear example, its discussion is not very effective.

The library's **chat** function provides the skeleton for a terminal-emulation

Like the Power Pack libraries reviewed earlier, PP3 is long on functions and short on documentation.

program—the user defines input and output translation tables and supplies functions to display help messages and data characters. A number of support functions set options for handling received characters. The **chat** function is a good idea, but its use is not obvious; better documentation is necessary. A short example in the manual, accompanied by a step-by-step discussion, would be a major improvement.

Library functions cover interrupt-driven and non-interrupt-driven communications, Hayes modem control, and the XMODEM protocol. A large number of functions are provided and in some respects the library is flexible and powerful. However, it lacks some important features, such as interrupt-driven output and support for Xon-Xoff flow-control messages received. XMODEM support functions do not allow access to individual data blocks in the application—they require the use of file I/O functions from the PP1 library. The most unfortunate limitation is the fixed 1KB buffer

for received characters; an ability to select the proper buffer size to fit an application is vital to good performance.

Several large sample programs are provided on the distribution diskettes. They illustrate use of the **chat** function to build a terminal-emulator program and the use of the XMODEM protocol. These programs are spread over a number of source files. The sample programs make heavy use of functions from other Power Pack libraries, so these programs cannot be recompiled (or relinked) without the other libraries. The source files for the sample programs contain warnings that other libraries are needed, but they are not specific as to which libraries.

The version of the Power Pack 3 Library tested contained some major errors. As a result, the library functions did not work properly with the COM2: port; the XMODEM receive functions also worked incorrectly.

One error causes the library functions to use the IRQ4 interrupt level instead of IRQ3 for COM2:. This meant that characters were not received when COM2: was used. Because of another error, the XMODEM support functions could not receive a file. The file could not be created because an argument was declared as **char path** rather than as **char *path**. Yet another error in the same XMODEM protocol caused received data blocks never to be written to the file.

An additional bug in the XMODEM support invalidated the checking of block numbers in received blocks once 32KB of data had been received—a positive acknowledgment was returned whether or not the block number was valid. (The 8-bit block number in the received data block was compared to a 16-bit expected block number maintained by the library functions.)

These bugs are neither subtle, nor minor—COM2: support does not work and the XMODEM support functions do not receive files. The bugs mentioned here were fixed in an updated version of the library, but because the same bugs were present in an earlier version (5.3), it appears that at least two versions of the library were shipped without proper testing. The version reviewed in this article failed the XMODEM transfer tests because of these bugs. A revised version from Software Horizons fixed these bugs but failed the test because of other bugs.

The Power Pack library is more complicated to use than either the Blaise or Greenleaf libraries, and its manual is less useful. Although the library has some unique features, it also lacks features in important areas. In addition, the presence of major bugs suggests that Software Horizons needs to improve its testing procedures.

None of the libraries reviewed is an unqualified success. In terms of the functions available, performance, and quality of the implementation (the source code), these products do not represent first-rate work.

Documentation is a weak point with each product. None provides the practical information that a programmer will need if he is to use them to solve a communications problem. The Blaise library includes competent documentation of its functions, although its manual is not up to the standards of other Blaise products reviewed in parts 1 and 2. The manual for the Greenleaf Library is out of date and short on information. XMODEM support functions do not work as described in the manual. The Software Horizons library is more complicated to use than either of the other products and its manual provides even less help than theirs.

DROP-IN MODULES

The Blaise library provides good support for the basic operations of sending and receiving data. The functions for receiving and transmitting data use interrupts, and Xon-Xoff flow control is well integrated with the interrupt functions. The Greenleaf library did not transmit data correctly and some of its functions are poorly designed. The Software Horizons library supports interrupts only for receiving data; also, its fixed 1KB buffer size is unsatisfactory.

All of the libraries provided some support for controlling a Hayes modem (or a Hayes-compatible modem). However, they all lacked practical information on using the library functions. They rely on the user to read the Hayes manual. If that is acceptable, then writing functions to send commands to a Hayes modem and collect responses from it should not be difficult.

None of the three provides a good implementation of the XMODEM protocol. The Blaise library functions work properly and are easy to use, but they are inefficient. The other libraries are inefficient, too, and they also contained bugs in the XMODEM support. The XMODEM support in all of these products is difficult to learn and to modify. Unfortunately, the packages do not provide a promising starting point for implementing any other protocol, either.

The Blaise library can be recommended for its support for basic character transmission and reception. Neither of the other two libraries can be recommended at present.

Async Manager 1.0: $175
Blaise Computing, Inc.
2034 Blake Street
Berkeley, CA 94704
415/540-5441
CIRCLE 357 ON READER SERVICE CARD

Greenleaf Communications Library 1.2: $160
Greenleaf Software, Inc.
2102 Hickory Drive
Carrollton, TX 75006
214/446-8641
CIRCLE 358 ON READER SERVICE CARD

Power Pack 3 (5.4): $149
Software Horizons, Inc.
165 Bedford Street
Burlington, MA 01803
617/273-4711
CIRCLE 359 ON READER SERVICE CARD

William J. Hunt is writing a book on software tools and C programs that illustrate real-world functions. Titled The C Toolbox, *the book is sheduled to be published later this year by Addison-Wesley.*

PERFORMANCE PACKAGE

Blaise Computing Inc. introduces the PERFORMANCE PACKAGE™ for Turbo Pascal programmers.

TURBO PASCAL

Turbo ASYNCH™

With Turbo ASYNCH, you can be in constant touch with the world without ever leaving the console. Rapid transit at its best. Turbo ASYNCH is designed to let you incorporate asynchronous communication capabilities into your Turbo Pascal application programs, and it will drive any asynchronous device via the RS232 ports, like printers, plotters, modems or even other computers. Turbo ASYNCH is fast, accurate and lives up to its specs. Features include…

◆ Initialization of the COM ports allowing you to set all transmission options. ◆ Interrupt processing. ◆ Data transfer between circular queues and communications ports. ◆ Simultaneous buffered input and output to both COM ports. ◆ Transmission speeds up to 9600 Baud. ◆ Input and output queues as large as you wish. ◆ XON/XOFF protocol.

The underlying functions of Turbo ASYNCH are carefully crafted in assembler for efficiency, and drive the UART and programmable interrupt controller chips directly. These functions, installed as a runtime resident system, require just 3.2K bytes. The interface to the assembler routines is written in Turbo Pascal.

The Turbo Pascal PERFORMANCE PACKAGE™ is for the serious Turbo Pascal programmer who wants quality tools to develop applications. Every system comes with a comprehensive User Reference Manual, all source code and useful sample programs. They require an IBM PC or compatible, utilizing MS-DOS version 2.0 or later. There are no royalties for incorporating PERFORMANCE PACKAGE functions into your applications.

Turbo POWER TOOLS and Turbo ASYNCH sell for $99.95 each, and they may be ordered directly from **Blaise Computing, Inc. TO ORDER, call (415) 540-5441**

NEW!

Turbo POWER TOOLS™

Turbo POWER TOOLS is a sleek new series of procedures designed specifically to complement Turbo Pascal on IBM and compatible computers. Every component in Turbo POWER TOOLS is precision engineered to give you fluid and responsive handling, with all the options you need packed into its clean lines. High performance and full instrumentation, including…

◆ Extensive string handling to complement the powerful Turbo Pascal functions. ◆ Screen support and window management, giving you fast direct access to the screen without using BIOS calls. ◆ Access to BIOS and DOS services, including DOS 3.0 and the IBM AT. ◆ Full program control by allowing you to execute any other program from within your Turbo Pascal application. ◆ Interrupt service routines written entirely in Turbo Pascal. Assembly code is not required even to service hardware interrupts like the keyboard or clock.

Using Turbo POWER TOOLS, you can now "filter" the keyboard or even DOS, and create your own "sidekickable" applications.

YES, send me the Best for the Best! Enclosed is ______ for ☐ Turbo ASYNCH ☐ Turbo POWER TOOLS. (CA residents add 6½% Sales Tax. All orders add $6.00 for shipping.)

Name: ______ **Phone:** ______

Shipping Address: ______ **Zip:** ______

City: ______ **State:** ______

VISA or MC #: ______ **Exp. Date:** ______

BLAISE COMPUTING INC.

BLAISE *watch us!*

◆ 2034 BLAKE STREET
◆ BERKELEY, CA 94704
◆ (415) 540-5441

LISTING 1: XMTBL.C

```
/* xmtbl.c - transmit chars test - Blaise version */
#include "stdio.h"
#include "asynch_1.h"

#define CARD  2

char combuf[1150] ;
int nerr = 0 ;
FILE *fopen() ;

main(argc,argv)
 int argc ;
 char *argv[] ;
 {
    int c , err , n , t , speed ;
    char b ;
    unsigned status ;
    FILE *in ;

    if( argc < 3 )
      { printf(" no file name on command line \n");
        exit(5) ;
      }

    in = fopen(argv[1],"rb");
    if( in == NULL )
      { printf(" can't open input file \n") ;
        exit(10) ;
      }
    sscanf(argv[2],"%d",&speed) ;
    init_a1(COM2,speed,2,0,2,&status,&status) ;
    err = open_a1(COM2,1000,100,0,0,combuf) ;/* set up for RS-232 use */

    send_chr('S') ;
    n = 0 ;
    eltime() ;/* start timing */
    while( (c = fgetc(in) ) != EOF )
      { send_chr(c) ; }
    send_chr('Q') ;
    t = eltime() ;/* stop timing */
    printf(" %d Ticks  %8.2f Secs \n",t,( (float) t)/18.2) ;

    fclose(in) ;
    wait_a1(100) ;
    close_a1(COM2) ;
 }


int rcv_chr()/* wait for and get next char */
 {
    int c , err , n ;
    char b ;
    unsigned status ;

    while( (err=rdch_a1(COM2,&b,&n,&status)) != 0)
      { ; }
    return( (int) b ) ;
 }


int send_chr(c)/* wait and xmt next char */
 int c ;
 {

    while( wrtch_a1(COM2,c)  != 0 )
      { ; }
 }
```

LISTING 2: RCVBL.C

```
/* rcvbl.c - receive chars test - Blaise version */
#include "stdio.h"
#include "asynch_1.h"

#define CARD  2

char combuf[1150] ;
int nerr = 0 ;
FILE *fopen() ;

main(argc,argv)
 int argc ;
 char *argv[] ;
 {
    int c , err , n , t , speed ;
    char b ;
    unsigned status ;
    FILE *out ;

    if( argc < 3 )
      { printf(" no file name on command line \n");
        exit(5) ;
      }
    out = fopen(argv[1],"wb");
    if( out == NULL )
      { printf(" can't open output file \n") ;
        exit(10) ;
      }
    scanf("%d",&speed) ;
    init_a1(COM2,speed,2,0,2,&status,&status) ;
    err = open_a1(COM2,1000,100,0,0,combuf) ;/* set up for RS-232 use */

    while( (c = rcv_chr() ) != 'S' )/* look for 'S' to start */
      { /* putchar(c) ; */ }
    eltime() ;/* start timing */
    c = rcv_chr() ;
    while( c != 'Q' )/* stop when 'Q' received */
      { fputc(c,out) ;/* put each char into a file */
        c = rcv_chr() ;
      }

    t = eltime() ;/* stop timing */
    printf(" %d Ticks  %8.2f Secs \n",t,( (float) t)/18.2) ;

    fclose(out) ;
    close_a1(COM2) ;
 }


int rcv_chr()/* wait for and get next char */
 {
    int c , err , n ;
    char b ;
    unsigned status ;

    while( (err=rdch_a1(COM2,&b,&n,&status)) != 0)
      { ; }
    return( (int) b ) ;
 }
```

LISTING 3: XMSGLF1.C

```
/* xmsglf1.c - Xmodem send using G'Leaf Comm lib */
#include "stdio.h"
#include "gf.h"
#include "asiports.h"

#define  CARD  2/* which RS-232 port to use */

char combuf[1150] ;
char obuf[1150] ;
int port = 1 ;
```

```
main(argc,argv)
 int argc ;
 char *argv[] ;
 {
    int c , err , n , t ,speed , lstat , mstat ;

    if( argc < 3 )
      { printf(" no file name or baud rate \n");
        exit(5) ;
      }

    sscanf(argv[2],"%d",&speed) ;
    err = asisetup(port,ASINOUT,0,combuf,1000,obuf,1000) ;
 printf(" asisetup - err=%d \n",err) ;
    asdtr(port,ON) ;
    err = asiinit(port,speed,0,1,8) ;
printf(" asiinit - err=%d \n",err) ;
    err = asistart(port,ASINOUT) ;
printf(" asistart - err=%d \n",err) ;
    eltime() ;
    err = send_filet(argv[1],&n) ;
    t = eltime() ;
    printf("  %4d Ticks  or %8.2f Secs \n",t,((float) t) / 18.2) ;
    printf("  send_file - err=%d     no recs. xmtted=%d \n",err,n) ;

    asiquit(port) ;
 }
int send_file(fn,pn)
 char fn[] ;/* file name */
 int *pn ;/* put record count here */
 {
    FILE *fd ;
    char buf[140] ;
    int block , ret , nr ;
    struct XMBUF locparms ;

    fd = fopen(fn,"rb") ;
    if( fd == NULL )
      { printf(" can't open file \n");
           return(-1) ;
      }

    block = 1 ;
    ret = -200 ;
    nr = fread(buf,1,128,fd) ;
    while( nr > 0 )
      { while( nr < 128 )
          { buf[nr] = '\0' ;
            nr ++ ;
          }
        ret = asixsendbuf(port,buf,block) ;
        if( ret == 0 )
            printf(" block %4d - sent OK \n",block );
        else
          { printf(" block %4d - error %d \n",block,ret);
            break ;
          }
        nr = fread(buf,1,128,fd) ;
        block++ ;
      }
    asiputc(port,EOT) ;
    fclose(fd) ;
    *pn = block ;
    return( ret ) ;
 }
```

LISTING 4: XMRGLF1.C

```
/* xmrglf1.c - Xmodem receive using G'Leaf Comm lib */
#include "stdio.h"
#include "gf.h"
#include "asiports.h"

#define  CARD  2/* which RS-232 port to use */

char combuf[1150] ;
```

```
char obuf[1150] ;
int port = 1 ;


main(argc,argv)
 int argc ;
 char *argv[] ;
 {
    int c , err , n , t ,speed , lstat , mstat ;

    if( argc < 3 )
      { printf(" no file name or baud rate \n");
        exit(5) ;
      }

    sscanf(argv[2],"%d",&speed) ;
    asisetup(port,ASINOUT,0,combuf,1000,obuf,1000) ;
    asdtr(port,ON) ;
    asiinit(port,speed,0,1,8) ;
    asistart(port,ASINOUT) ;

    eltime() ;
    err = rcv_filet(argv[1],&n) ;
    t = eltime() ;
    printf("  %4d Ticks  or %8.2f Secs \n",t,((float) t) / 18.2) ;
    printf("  rcv_file - err=%d     no recs. xmtted=%d \n",err,n) ;

    asiquit(port) ;
 }

int rcv_file(fn,pn)

 char fn[] ;/* file name */
 int *pn ;/* put record count here */
 {
    FILE *fd ;
    char buf[140] ;
    int block , ret ,nw ;
    struct XMBUF locparms ;

    fd = fopen(fn,"wb") ;
    if( fd == NULL )
      { printf(" can't open file \n");
        return(-1) ;
      }

    block = 1 ;
    ret = -200 ;
    while( 1 == 1 )
      { locparms.xblocknum = block ;
        ret = asixrecvbuf(port,buf,&locparms) ;
        if( ret == 0 )
          {
            nw = fwrite(buf,1,128,fd) ;
            printf(" block %4d - rcvd OK \n",block );
          }·
        else if( locparms.xstatus == 6 )
          { printf(" rcvd EOT \n") ;
            asiputc(port,ACK) ;
            break ;
          }
        else
          { printf(" block %4d - error %d \n",block);
            break ;
          }
        block ++ ;
      }
    fclose(fd) ;
    *pn = block ;
    return( ret ) ;
 }
```

“I waste a lot of time contacting vendors and then find out they don’t have what I need.”

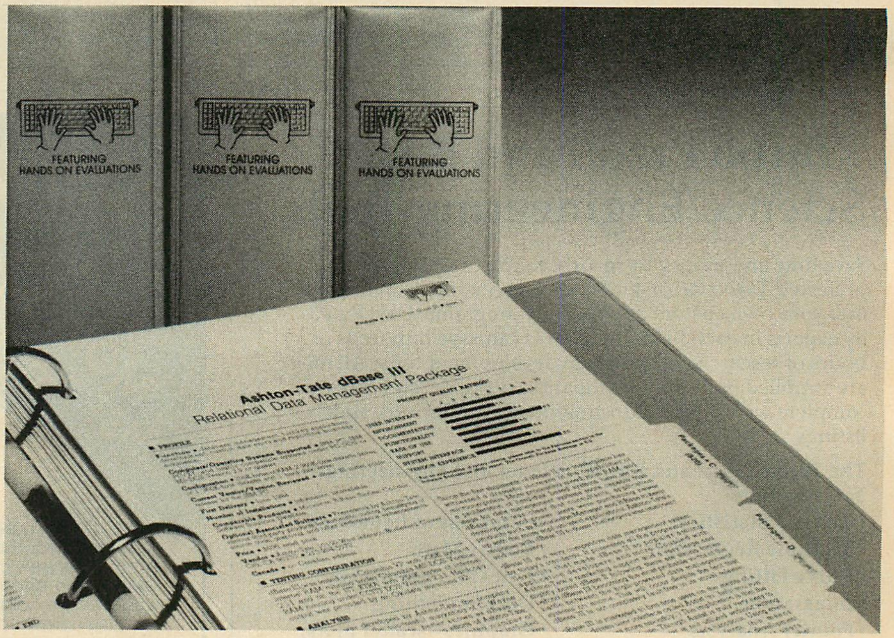
“Suddenly I’m the corporate guru on micros, but I don’t have answers to the questions I’m being asked.”

“I’m already spending too much time on product evaluations, and the field gets bigger every day.”

—If these complaints sound familiar, this is the help you need.

INTRODUCING DATA DECISIONS MICROCOMPUTERS

An ALL-NEW information service for the corporate micro buyer that features HANDS-ON TESTING of micro software and peripherals by experts in Data Decisions’ own labs.



That’s right, now you can get the results of hundreds of hands-on tests done under consistent, controlled conditions by our specially trained team of micro experts—who will tell you what each product’s strength and limitations are, how competing products compare, and what you can expect when you bring them in-house.

And hands-on evaluations of software and peripherals are just the beginning! Data Decisions MICROCOMPUTERS is your one-step look-up source for what you need to know about LANs... PC-compatibles... specialized workstations—for ALL your micro information needs.

MICROCOMPUTERS comes to you in 3 loose-leaf volumes which are updated every month to ensure their accuracy. They give you:

- Hands-on evaluations of key software and peripheral products.
- Rating charts that let you compare how competing packages measure up in actual use.
- In-depth analyses that highlight strengths and limitations of mainstream hardware, software and peripheral products.
- Action-oriented technology reports that fill you in on issues and trends.
- Broadbased surveys on accounting software, electronic mail packages and other important areas.
- Lively monthly newsletters that bring you late-breaking industry happenings and new product announcements, point out the best articles published in recent computer magazines, and provide a summary of the month’s hands-on testing scores and a calendar of industry events.
- Unlimited free telephone consulting by micro experts whose only job is to answer your questions.

Whether you’re looking for 10 micros or 100, an integrated software package, a micro-to-mainframe down-loader, graphics—or the right combination of products capable of working together, MICROCOMPUTERS has the information you need.

Data Decisions MICROCOMPUTERS covers the market from A to Z.

Microcomputers: desktops, intelligent terminals, UNIX systems, execu-

tive workstations, integrated office automation workstations, supermicros, kneetops, portables, transportables...AND MORE.

Software: graphics, accounting packages, financial packages, training packages, operating systems, utilities, DBMSs, office automation, word proessing, communications—PLUS the new integrated packages...AND MORE.

Expansions/Extension Cards: controllers, clocks, A to D converters, extra quad boards, interface boards, language cards, solid state disks, memory boards, processor enhancement, auxiliary processors...AND MORE.

Communications: micro LAN networks, micro modems, micro terminals, terminal emulators, micro-to-mainframe downloaders, database access packages, electronic mail... AND MORE.

Peripherals/Accessories/Support Equipment: printers, plotters, monitors, floppies, hard disks, keyboards, tape drives, OCR bar codes, sensors, power supplies, power surge protectors, cooling fans...AND MORE.

"Just one report saved me three nights of tinkering at a computer store on my own time. It's like having a consultant sitting right there at the end of the desk."

—Bob Butler, Mechanical Engineer
Gillette Company Boston, MA

Data Decisions MICROCOMPUTERS service is:

Comprehensive: It offers you the most complete, concise, accurate information ever compiled on micros.

Authoritative: It's put together by the largest, most experienced staff of micro experts in the publishing industry today —experts who don't just read about products, but who test them in our labs.

Analytic: Hard-hitting independent analysis highlights strengths, limitation and suitability to a corporate environment of all mainstream products.

Practical: Reliable user ratings let you see how products measure up in actual use.

Convenient: One service gives you all the information you need.

"Your telephone consultants are just outstanding—a really great group. They provide timely, personal response to virtually any type of inquiry."

—Stan Leu
Director of Marketing, DS Agency Inc.
(Agents for Exxon Office Systems)
Lubbock, TX

Economical: A 1-year subscription costs no more than you would have to pay a consultant to do just one of our hands-on evaluations.

Easy-to-Use: Specially designed dot charts point you quickly to the products that meet your needs...Rating bar charts let you see at a glance how similar products compare...Easy-to-read graphics point out the differences between models from the same vendor... Consistent formatting makes it easy to compare competitive offerings and find specific facts.

Current: Up-to-the-minute newsletters and report updates reach you every month...Complete revision of all product reports are issued at least once a year.

Personalized: Each MICROCOMPUTERS subscriber is eligible for unlimited free telephone consulting with microcomputer experts whose only job is to provide quick answers to your questions.

"I looked at two other micro services but yours was far superior. Your people do a wonderful job on the hands-on evaluations. I particularly appreciate the way you look at a product from a corporate standpoint, not just from the technical point of view.

I had MICROCOMPUTERS at my last job, and when I came here, ordering a new subscription was one of the first things I did."

—Gerald Nadeau, Systems Analyst
Analog Devices, Inc. Burlington, MA

have used Data Decisions MICROCOMPUTERS to help them make the right choices.

The annual subscription rate—complete with all 3 volumes, updates, newsletters and free unlimited phone consulting service—is only $900 plus $25 shipping and handling... far less than you'd have to pay an outside consultant for even one of our in-depth evaluations.

Data Decisions is so confident of your response to MICROCOMPUTERS that we are offering to let you examine all three volumes at your leisure in your own office with absolutely no purchase obligation. The only charge for this outstanding offer is the shipping and handling fee—which will be credited to your purchase if you subscribe.

Order your trial review today and see for yourself how Data Decisions MICROCOMPUTERS can save you hundreds of hours of product testing and thousands of dollars in costly purchasing mistakes.

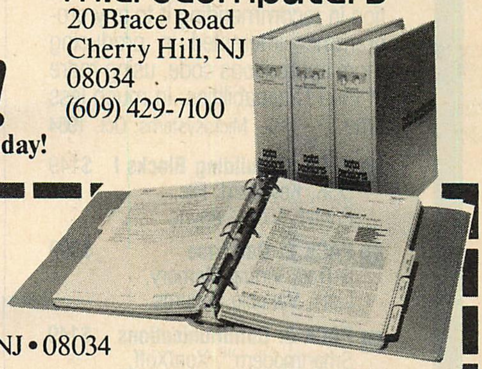
SPECIAL NO-PURCHASE-OBLIGATION TRIAL REVIEW OFFER

Thousands of smart DP professionals and micro buyers like yourself

ACT NOW!

Complete this order form and mail today!

Data Decisions microcomputers
20 Brace Road
Cherry Hill, NJ
08034
(609) 429-7100



Data Decisions microcomputers
20 Brace Road • Cherry Hill, NJ • 08034

☐ **YES!** Please send me Data Decisions MICROCOMPUTERS for a no-purchase-obligation trial review. I understand that the only charge is a $25 shipping/handling fee which will be credited to my purchase if I subscribe.
C5

☐ Although your offer sounds great, I'm still not convinced. Please send me additional information.
Y4

Signature ______________________

Name ______________ Title ______________

Company ______________________

Address ______________________

City ______________ State ______ Zip ______

Phone ( )______________________

A200

YOU NEED A GOOD LIBRARY

POWER PACKS™

COMPLETE SOURCES NO ROYALTIES

COMPREHENSIVE C Power Packs include over 1000 functions which provide an integrated environment for developing your applications efficiently. "This is a beautifully documented, incredibly comprehensive set of C Function Libraries."
— Dr. Dobb's Journal, July 1984

USEFUL "...can be used as an excellent learning tool for beginning C Programmers..."
– PC User's Group of Colorado, Jan. 1985

FLEXIBLE Most Compilers and all Memory Models supported.

RECOMMENDED "I have no hesitation in recommending it to any programmer interested in producing more applications code, using more of the PC capabilities, in much less time." — Microsystems, Oct. 1984

- **PACK 1: Building Blocks I** $149
  DOS, Keyboard, File, Printer, Video, Async
- **PACK 2: Database** $399
  B-Tree, Virtual Memory, Lists, Variable Records
- **PACK 3: Communications** $149
  Smartmodem™, Xon/Xoff, X-Modem, Modem-7
- **PACK 4: Building Blocks II** $149
  Dates, Textwindows, Menus, Data Compression, Graphics
- **PACK 5: Mathematics I** $99
  Log, Trig, Random, Std Deviation
- **PACK 6: Utilities I (EXE files)** $99
  Arc, Diff, Replace, Scan, Wipe

Master Card/Visa, $7 Shipping, Mass. Sales Tax 5%

ASK FOR FREE DEMO DISKETTE

NOVUM ORGANUM INC. SOFTWARE HORIZONS INC.

165 Bedford St., Burlington, MA 01803
(617) 273-4711

CIRCLE NO. 223 ON READER SERVICE CARD

PROLOG-86™

Become Familiar in One Evening

Thorough tutorials are designed to help learn the PROLOG language quickly. The interactive PROLOG-86 Interpreter gives immediate feedback. In a few hours you will begin to feel comfortable with it. In a few days you are likely to know enough to modify some of the more sophisticated sample programs.

Sample Programs are Included like:

- **an EXPERT SYSTEM**
- **a NATURAL LANGUAGE INTERFACE**
  (it generates a dBASEII "DISPLAY" command)
- **a GAME** (it takes less than 1 page of PROLOG-86)

PROTOTYPE Ideas and Applications QUICKLY

Serious development of experimental systems and prototypes is practical with the full syntax of PROLOG-86. 1 or 2 pages of PROLOG is often like 10 pages in "C".

Programming Experience is not required but a logical mind is. PROLOG-86 supports the de facto STANDARD.

RECENT IMPROVEMENTS: Access to MSDOS, on-line help, load Editor.

AVAILABILITY: All MSDOS, PCDOS systems.

FREE with order: "Best of Prolog-86 Programs"—contest entries include: a primate expert system, an automobile expert system, a blocks world natural language system, etc. Call before November 30.

Only $125.
Full refund if not satisfied during first 30 days.

Solution Systems™

335- P Washington St.,
Norwell, Mass. 02061
617-659-1571
800-821-2492

CIRCLE NO. 220 ON READER SERVICE CARD

Now Supports Microsoft 3.0

$149.95

Vitamin C

Windows that open, close, grow, shrink, move & scroll. Input validation,formatting, editing & processing. Help messages by field, by key word & from help file. Date & time math, attribute control, pull down menus, and more!

Now even your simplest applications can easily include features that you didn't used to have time or patience enough to tackle. Vitamin C is more than a library full of building blocks. It is a well planned, tightly woven set of high level functions for quick results PLUS low level routines for complete control. Complete source code, no royalties. Great manual with tutorial and reference. Sample programs too! For Microsoft 3.0, CI-C86, & Lattice C. Call for other versions, systems & products.

CALL TODAY!
Or send 149.95+ $3 shipping and handling. Texas add sales tax. MasterCard and Visa accepted.

Creative Programming
Box 112097
Carrollton, Tx 75011
(214)783-9388

Extended Technical support available.

CIRCLE NO. 124 ON READER SERVICE CARD

GUY QUEDENS and GARY WEBB

Switching Modes

Getting into and out of the iAPX 286's protected virtual mode helps take full advantage of the AT's power.

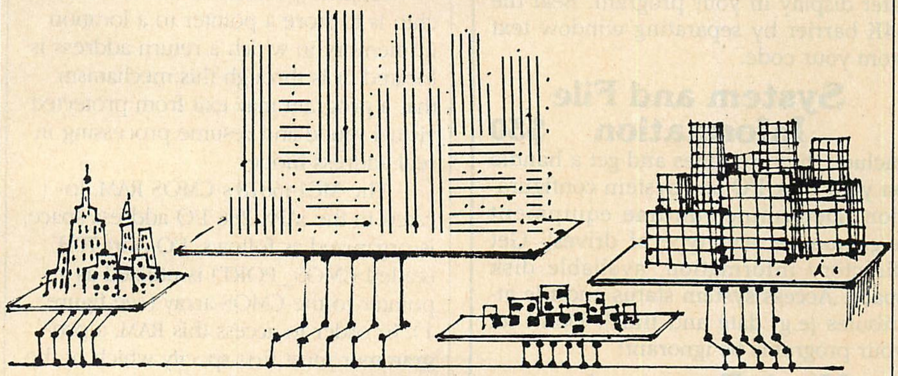


ILLUSTRATION • MACIEK ALBRECHT

In computer architectures, power always has its intellectual price. Writing assembly language programs for the 8086/88 is difficult enough; the introduction of the 80286-based PC/AT, with its advanced architectural features, has made the task even more complicated.

The 80286 architecture is the first in its series to perform memory management on the CPU chip itself. Prior to the 286, all 86-family microprocessors operated in real-address mode, in which the address specified in a memory reference instruction was the actual physical address in the computer's memory system. In addition to real-address mode, the 80286 also has a protected virtual mode, in which the address specified in a memory reference instruction is only a virtual address, with no independent relationship to machine memory. The virtual address is translated by the chip into a physical address in machine memory. Protected virtual mode employs a protection scheme that includes hardware-enforced checking of length and type to protect memory segments from deliberate or inadvertent misuse.

There are several methods for getting into and out of protected virtual mode. The following explanations of these methods require a working understanding of the architecture of the iAPX 286, its instruction set, and its addressing schemes. To gain such as understanding, review "The Evolution of the iAPX 286" by Bob Greene (*PC Tech Journal*, December 1984, page 118).

Initially, the iAPX 286 CPU that is used in the PC/AT utilizes real-address mode. Issuing a processor RESET (which is done by raising to a logical 1 a physical pin on the iAPX 286 chip) forces several of its registers into a predetermined state (see table 1). This is the state of the AT upon power-up. The least significant bit of the machine status word is set to 0, indicating that the iAPX 286 is in real-address mode. From real-address mode, it is possible to move to protected virtual mode using one of the methods described later.

To return to real-address mode from protected virtual mode, a *shutdown* may be invoked; this shutdown causes a RESET, which then forces the iAPX 286 to return to real-address mode. To invoke a shutdown, execute this sequence of instructions:

```
    MOV AL, SHUT_CMD
    OUT STATUS_PORT,AL
L1: HLT
    JMP L1
```

(Note in listing 1 that SHUT_CMD and STATUS_PORT have been previously equated to FEH and 64H, respectively.)

STATUS_PORT is the port address of a chip (called an 8042), which is a general-purpose, Universal Peripheral Interface, 8-bit microcomputer. The 8042 is a low-cost microcomputer with 2KB of program memory, 128 bytes of data memory, and 8-bit CPU, I/O ports, an 8-bit timer/counter, and a clock generator—all in a single 40-pin package. The status of the 8042 may be read from the iAPX 286 through an I/O port. STATUS_PORT contains eight single-bit fields, which are named and briefly described on page 5-27 of the *PC/AT Technical Reference.* (Some of these bit fields are mentioned later.)

The 8042's responsibilities include control of the keyboard and speaker interface on the AT. Sheet 17 of the *PC/AT Technical Reference* illustrates the 8042 and its interface circuitry. Note that 8042 pin 21, called RC, ties into a flip-flop on sheet 22, which further ties into a line called SHT DWN on sheet 1. SHT DWN is tied to the RESET pin on the iAPX 286. Using this hardware path, the 8042 can reset the 286.

SHUT_CMD, once accepted by the 8042, results in its sending a RESET signal to the iAPX 286. The halt (HLT) stops the processor, giving the 8042 time to initiate a RESET. The JMP L1 insures the HLT against stray interrupts, which can restart the processor from a halted state. If an unwanted interrupt restarts the processor, control immediately jumps back to the HLT instruction.

When the AT is powered up, hardware circuits force a RESET. As the BIOS is initialized, it looks at STATUS_PORT. Bit field D2 of STATUS_PORT is named SYS_FLAG in the AT ROM BIOS listings. If SYS_FLAG = 0 (as it does on power-up), the BIOS recognizes this as a hard reset and proceeds with total system initialization. Conversely, if SYS_FLAG = 1 (a result of issuing SHUT_CMD), the BIOS recognizes that the RESET was caused by a shutdown.

Once the BIOS knows the RESET came from software via shut down, it accesses another hardware device to determine the next processing steps to take. This device is an MC146818 realtime clock/complementary metal oxide semiconductor (RT/CMOS) RAM chip that contains a realtime clock and 64 bytes of CMOS RAM. The clock function uses 14 bytes of this RAM; the rest is allocated to maintain system configuration: disk type(s), installed equipment flags, memory sizes. Its contents are maintained through power-down times by a battery, and it is outside of normal system memory space.

Another purpose of this memory chip is to store a pointer to a location in memory in which a return address is located. It is through this mechanism that a program may exit from protected virtual mode and resume processing in real-address mode.

The MC146818's CMOS RAM, located in the iAPX 286 I/O address space, is organized as follows: I/O port 70H (called CMOS_PORT) is utilized as a pointer to the CMOS array (see figure 1). In order to access this RAM, a programmer must first specify which of the 64 bytes he wants; this is accomplished by executing an OUT instruction to CMOS_PORT, which outputs the array index to the MC146818.

To read or write the CMOS RAM, execute an IN or OUT to port 71H (CMOS_PORT+1). An IN instruction reads from the array element specified by the OUT 70H instruction as described above; an OUT instruction writes to the array element specified by the OUT 70H instruction. Notice that at any one time only one byte of the CMOS RAM in the MC146818 is ever directly accessed by the iAPX 286.

Once the BIOS has determined that the RESET was invoked by a shutdown, it accesses the CMOS shutdown status byte (index 0FH). This byte contains a value from 00H to 0AH, which, when multiplied by 2, is used as an index into a branching table. This table is in the AT ROM BIOS. (It is shown in the ROM BIOS listings beginning at the BRANCH: label on page 5-34 of the *PC/AT Technical Reference.*) Control is then transferred to whatever address is at that computed index in the table. It is through this table that a program regains control in real-address mode.

The following two methods of entering and leaving protected virtual mode operate completely differently.

Method 1, the generally preferred method, uses the shutdown status byte index values 05H and 0AH, which constitute a mechanism provided for by the BIOS for returning to real-address mode. When utilizing this facility, however, the programmer must keep in mind that when the BIOS reboots the machine (as a result of a RESET), the two 8259 interrupt-controller chips are reinitialized. Thus, the programmer must reset these chips himself; in addition, he must remember that any pending interrupts will be canceled. The difference between 05H and 0AH is that 05H will cause the keyboard buffer and the last timer tic to be flushed.

Method 2 uses shut-down status byte value 09, which involves the reentry code of the BIOS Block Move function. Block Move (INT 15H when AH=87H) provides a method of moving a block of data to and from extended memory using real-address mode, without the application having to be concerned with protected virtual mode.

Method 2 does not actually call the Block Move interrupt, but simply makes use of some of the code Block Move uses to clean house when it returns to real-address mode. Therefore, the programmer must simulate an interrupt by pushing the environment and return address on the stack before actually entering protected virtual mode. Then when Block Move's return code is used, it will look like an interrupt, and all the information the return code requires to effect a successful return to real-address mode will be provided.

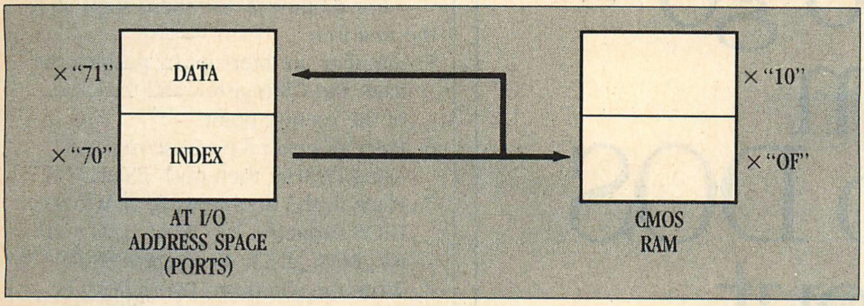
If method 2 is used, the reinitialization of the 8259 interrupt-controller

TABLE 1: *Processor RESET*

REGISTER	CONTENTS
FLAGS	0002H
MSW	FFF0H
IP	FFF0H
CS Selector	F000H
DS Selector	0000H
SS Selector	0000H
ES Selector	0000H
CS Base	FF0000H
DS Base	000000H
SS Base	000000H
ES Base	000000H
CS Limit	FFFFH
DS Limit	FFFFH
SS Limit	FFFFH
ES Limit	FFFFH
IDT Base	000000H
IDT Limit	FFFFH

Issuing a processor RESET (done by raising to a logical 1 a physical pin on the iAPX286 chip) forces several of its registers into a predetermined state.

FIGURE 1: *How CMOS RAM Is Accessed*



In order to access CMOS RAM, a programmer must first specify which of the 64 bytes he wants to access; this is accomplished by executing an OUT instruction to CMOS_PORT, which outputs the array index to the MC146818.

chips is bypassed. This constitutes something of a back-door approach and is not guaranteed to work in subsequent releases of the PC/AT BIOS; specifically, if IBM changes the logic of the reentry code—that is, the order in which the registers are restored from the stack—this second method may no longer work properly.

An important point concerning the transition from real-address to protected virtual mode should be made: while load machine status word (LMSW) is executed, the next instruction is loaded by the processor's Bus Interface Unit (BIU) into a prefetch queue, ready to be executed when LMSW completes. When LMSW completes, however, CS is an *index* into a descriptor table, not a physical segment pointer. CS:IP now points to an unknown spot and at an unknown code. Fortunately, however, the BIU already loaded the next instruction from CS:IP when CS was a REAL segment pointer. This instruction *must* be a FAR jump to "segment:offset," which changes the CS register, thereby purging the processor's prefetch queue and forcing it to get the next instruction from the new CS:IP.

Listings 1 and 2 illustrate the techniques of methods 1 and 2, respectively. The first four steps are the same for both methods:

1. Define a global descriptor table (GDT) and/or a local descriptor table (LDT), load with appropriate values.
2. Load the global descriptor table register (GDTR) and/or the local descriptor table register (LDTR).
3. To allow interrupts in protected virtual mode, define and load the interrupt descriptor table (IDT) and load the interrupt descriptor table register (IDTR).
4. Gate address line A20 active. This should be done before entering protected virtual mode; the line should be gated off inactive upon reentering real-address mode.

The last of these actions bears some explaining. There are, of course, 24 memory address lines in the AT, called A0 through A23. In most computers, these lines go straight from the processor to the bus. In the AT, however, an unexplained hardware gating switch exists on line A20 between the iAPX 286 and the bus. (See flip-flop F257 on sheet 1, p. 1-61 of the *PC/AT Technical Reference.*) If this gate is shut, it will place a 0 on A20 irrespective of what the iAPX 286 originally places on A20. A 0 on A20 causes writes (and reads) to 100000H to be directed to 000000H; those to 300000H will be directed to 200000H, and so on, in effect mapping all odd-megabyte (1, 3, 5, etc.) blocks of memory on top of the even-megabyte blocks preceding them. The gate must be open (active) for the odd-megabyte blocks to be accessible. IBM ordinarily leaves this gate open (allowing unimpeded access to the odd-numbered megabyte blocks) and has given no clue as to its purpose.

For method 1, the following steps would be used after the first four:

5. Save the status of the 8259 interrupt-controller chips.
6. Place the double-word (DWORD) return address in the BIOS variables IO_ROM_SEG and IO_ROM_INIT.
7. Write the shutdown byte to CMOS memory, using either 05H or 0AH.
8. Load machine status word (LMSW) and JUMPFAR into protected mode.

To exit protected virtual mode, returning to real-address mode, do the following:

9. Output the shutdown command to the status port.
10. Issue a halt (HLT) instruction. The program will resume at the address specified in step 6 above.
11. Degate address line A20.

12. Restore the 8259 interrupt-controller chips' status.

For method 2, use these steps after the first four:

5. Simulate an interrupt by pushing the flags, the CS register, and the offset of the reentry point.
6. Push all general-purpose registers using PUSHA, then push ES and DS.
7. Place in the double word (DWORD) BIOS variables, IO_ROM_SEG and IO_ROM_INIT, the current SS:SP.
8. Write the shutdown byte to CMOS memory. The value to use is 09H.
9. Load the machine status word (LMSW) and JUMPFAR into protected virtual mode.

To exit protected virtual mode, returning to real-address mode, follow these steps:

10. Output the shutdown command to the status port.
11. Issue a halt (HLT) instruction. The program will resume at the address specified in step 5 above, with the machine state just as it was prior to step 7. The BIOS Block Move reentry logic automatically degates address line A20.

As computers become more powerful and more complicated, it becomes more difficult to make use of the full measure of a machine's performance ability. These tools should help programmers exploit the AT's full potential.

REFERENCES

IBM Technical Reference Personal Computer AT. Boca Raton, FL: IBM Corporation, 1984.

Macro Assembler Version 2.0 Reference. Boca Raton, FL: IBM Corporation, 1984.

iAPX 286 Hardware Reference Manual. Santa Clara, CA: Intel Corporation, 1984.

iAPX 286 Programmer's Reference Manual including the iAPX 286 Numeric Supplement. Santa Clara, CA: Intel Corporation, 1984.

Microsystem Components Handbook. vol. 1. Santa Clara, CA: Intel Corporation, 1985.

Integrated Instrumentation and In-Circuit Emulation System Reference Manual. Santa Clara, CA: Intel Corporation, 1984.

Guy Quedens, a systems programmer for Quadram Corporation, holds a bachelor of science degree in mathematics and computer information systems from Troy State University in Alabama. Gary Webb, an independent consulting electrical engineer, heads his own firm, Tigerlab Corporation, which is based in Boca Raton, Florida.

LISTING 1: METHOD1.ASM

```
        .286C     ; Tell MASM 2.0 about 286 instructions
;---------------------------------------------------------------:
;       Sample Program 1                                        :
;                                                               :
; This program switches into Protected Virtual Mode, changes    :
; the display attribute to reverse video, and returns to Real   :
; Mode to exit to DOS                                           :
;                                                               :
; Once entered into a file, do the following:                   :
;       MASM SAMPLE1;                                           :
;       LINK SAMPLE1;                                           :
;       EXE2BIN SAMPLE1 SAMPLE1.COM                             :
;       DEL SAMPLE1.EXE                                         :
;                                                               :
; WARNING: This program will "kill" a PC.  I should only        :
; be run on an AT.                                              :
;---------------------------------------------------------------:

bios_data_seg SEGMENT at 0040h
              ORG       0067h
io_rom_init   dw ?      ; dword variable in BIOS data segment
io_rom_seg    dw ?      ;  used to store a dword address
bios_data_seg ENDS

descriptor    STRUC
seg_limit     dw 0      ; segment limit (1-65536 bytes)
base_lo_word  dw 0      ; 24 bit physical address
base_hi_byte  db 0      ; (0 - (16M-1))
access_rights db 0      ; access rights byte
              dw 0      ; reserved_386
descriptor    ENDS

cmos_port       equ 070h
code_seg_access equ 10011011b ;access rights byte for code seg
data_seg_access equ 10010011b ;access rights byte for data seg
disable_bit20   equ 11011101b ;8042 function code to de-gate A20
enable_bit20    equ 11011111b ;8042 function code to gate A20
inta01          equ 021h      ;8259 Int Controller #1
intb01          equ 0A1h      ;8259 Int Controller #2
port_a          equ 060h      ;8042 port A
shut_cmd        equ 0FEh      ;cmd to 8042: shut down AT
shut_down       equ 00Fh      ;CMOS shut down byte index
status_port     equ 064h      ;8042 status port
virtual_enable  equ 0001h     ;LSB=1: Protected Virtual Mode

        SUBTTL  Macro Definitions
        PAGE
;---------------------------------------------------------------:
; These mnemonics are not supported in MASM 2.0 therefore       :
; they are supplied here as MACROS.                             :
;---------------------------------------------------------------:
lgdt    MACRO   lgdt1     ;; Load Global Descriptor Table
        LOCAL   lgdt2,lgdt3
        db      0Fh
lgdt2   label   byte
        mov     dx,word ptr lgdt1
lgdt3   label   byte
        org     offset lgdt2
        db      001h
        org     offset lgdt3
        ENDM

lmsw    MACRO   lmsw1     ;; Load Machine Status Word
        LOCAL   lmsw2,lmsw3
        db      0Fh
lmsw2   label   byte
        mov     si,ax
lmsw3   label   byte
        org     offset lmsw2
        db      001h
        org     offset lmsw3
        ENDM

                          ;; This is a "hard coded" far jump
jumpfar MACRO   jumpfar1,jumpfar2
        db      0EAh
        dw      (offset jumpfar1)
        dw      jumpfar2
        ENDM

        SUBTTL    Program entry point and data area
        PAGE
cseg    SEGMENT para      public      'code'
        ASSUME  cs:cseg

        ORG     100h
start:  jmp     short     main

        EVEN
gdt     LABEL   word

gdt_desc    EQU  (($-gdt)/8)*8 + 0000000000000000b
gdt1        descriptor  <gdt_leng,,,data_seg_access,>
cs_code     EQU  (($-gdt)/8)*8 + 0000000000000000b
gdt2        descriptor  <cseg_leng,,,code_seg_access,>
cs_data     EQU  (($-gdt)/8)*8 + 0000000000000000b
gdt3        descriptor  <cseg_leng,,,data_seg_access,>
ss_desc     EQU  (($-gdt)/8)*8 + 0000000000000000b
gdt4        descriptor  <0FFFFh,,,data_seg_access,>
ds_desc     equ  (($-gdt)/8)*8 + 0000000000000000b
gdt5        descriptor  <0FFFFh,,,data_seg_access,>
es_desc     equ  (($-gdt)/8)*8 + 0000000000000000b
gdt6        descriptor  <0FFFFh,,,data_seg_access,>
gdt_leng    EQU  $-gdt
            PAGE
;---------------------------------------------------------------------:
; Format of the Segment Selector Component:                          :
;                                                                     :
; +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+                   :
; |        INDEX                          +TI+ RPL +                  :
; +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+                   :
;                                                                     :
; TI = Table Indicator (0=GDT, 1=LDT)                                 :
; RPL = Requested Privelege Level (00 = highest; 11 = Lowest)         :
;---------------------------------------------------------------------:
; Format of the Global Descriptor Table                               :
;                   .-----------+                       +---> TI      :
;                   V           |                       |++-> RPL     :
;      GDT ==> +--------------+ |                       |||           :
;              |   GDT_DESC   | --+   0000000000000000b               :
;              +--------------+                                       :
;              |   CS_CODE    |       0000000000001000b               :
;              +--------------+                                       :
;              |   CS_DATA    |       0000000000010000b               :
;              +--------------+                                       :
;              |   SS_DESC    |       0000000000011000b               :
;              +--------------+                                       :
;              |   DS_DESC    |       0000000000100000b               :
;              +--------------+                                       :
;              |   ES_DESC    |       0000000000101000b               :
;              +--------------+                                       :
;---------------------------------------------------------------------:

i8259_1 db      ?         ; store for status of 8259 #1
i8259_2 db      ?         ; store for status of 8259 #2

        SUBTTL Program Main
        PAGE
;---------------------------------------------------------------------:
; MAIN                                                                :
;---------------------------------------------------------------------:
        ASSUME  ds:cseg
main    PROC                          ;ES=DS=CS
        cld                           ;forward

        mov     dx,cs                 ;form 24bit address out of
        mov     cx,offset gdt         ; CS:GDT
        call    form_24bit_address
        mov     gdt1.base_lo_word,dx  ;DESC now points to gdt
        mov     gdt1.base_hi_byte,cl

        mov     dx,cs                 ;form 24bit address out of
        xor     cx,cx                 ;  CS:0000
        call    form_24bit_address
```

```
        mov     gdt2.base_lo_word,dx ;CS_CODE now points to
        mov     gdt2.base_hi_byte,cl ; CSEG as a code segment
        mov     gdt3.base_lo_word,dx ;CS_DATA now points to
        mov     gdt3.base_hi_byte,cl ; CSEG as a data segment

        mov     dx,ss           ;form 24bit address out of
        xor     cx,cx           ; SS:0000
        call    form_24bit_address
        mov     gdt4.base_lo_word,dx ;SS_DESC now points to
        mov     gdt4.base_hi_byte,cl ; stack segment

        lgdt    gdt             ;Load the GDTR

        mov     ah,enable_bit20 ;gate address bit 20 on
        call    gate_a20
        or      al,al           ; was the command accepted?
        jz      m_10            ; go if yes
        mov     dx,offset gate_failure ;print error msg
        mov     ah,9                   ; and terminate
        int     21h
        int     20h

gate_failure    db  "Address line A20 failed to Gate open$"

m_10:   cli                     ;No interrupts

        in      al,inta01  ;get status of Int Controller #1
        mov     i8259_1,al
        in      al,intb01  ;get status of Int Controller #2
        mov     i8259_2,al

        ASSUME  ds:bios_data_seg
        mov     dx,bios_data_seg ;Real Mode Return address
        mov     ds,dx
        mov     io_rom_seg,cs
        mov     io_rom_init,offset real

        mov     al,shut_down    ;Set shutdown byte
```

```
        out     cmos_port,al    ; to shut down x05.
        jmp     short  $+2      ;I/O delay
        mov     al,5
        out     cmos_port+1,al

        mov     ax,virtual_enable ;machine status word needed to
        lmsw    ax              ;switch to virtual mode
        jumpfar m_20,cs_code    ;Must purge prefetch queue

m_20:   ASSUME  ds:cseg         ;IN VIRTUAL MODE ...
        mov     ax,ss_desc      ;stack segment selector
        mov     ss,ax           ;user's ss+sp is not a descriptor

        mov     ax,cs_data
        mov     ds,ax           ;DS = CSEG as data

        mov     gdt5.base_lo_word,0000h  ;use 8000 for COLOR
        mov     gdt5.base_hi_byte,0Bh
        mov     gdt6.base_lo_word,0000h
        mov     gdt6.base_hi_byte,0Bh

        mov     ax,ds_desc
        mov     ds,ax
        mov     ax,es_desc
        mov     es,ax
        mov     cx,80*25
        xor     si,si
        xor     di,di
m_30:   lodsw
        mov     ah,70h          ;attribute reverse video
        stosw
        loop    m_30

        mov     al,shut_cmd     ;shutdown cmd
        out     status_port,al  ;get back into REAL mode
m_40:   hlt
        jmp     short m_40
```

```
        SUBTTL  Gate A20
        PAGE
;-----------------------------------------------------------:
; GATE_A20                                                  :
; This routine controls a signal which gates address bit 20. :
; The gate A20 signal is an output of the 8042 slave processor.:
; Address bit 20 should be gated on before entering protected :
; mode.  It should be gated off after entering real mode from :
; protected mode.                                           :
; Input:  (AH)=0DDh addr bit 20 gated off (A20 always 0)    :
;         (AH)=0DFh addr bit 20 gated on (286 controls A20) :
; Output: (AL)=0 operation successful.  8042 has accepted cmd :
;         (AL)=2 Failure -- 8042 unable to accept command.  :
;-----------------------------------------------------------:
gate_a20        PROC
        cli                     ;disable ints while using 8042
        call    empty_8042      ;insure 8042 input buffer empty
        jnz     gate_a20_01     ;ret if 8042 unable to accept cmd
        mov     al,0D1h         ;8042 command to write output port
        out     status_port,al  ;output cmd to 8042
        call    empty_8042      ;wait for 8042 to accept command
        jnz     gate_a20_01     ;ret if 8042 unable to accept cmd
        mov     al,ah           ;8042 port data
        out     port_a,al       ;output port data to 8042
        call    empty_8042      ;wait for 8042 to port data
gate_a20_01:
        ret
gate_a20        ENDP
;-----------------------------------------------------------:
; EMPTY_8042                                                :
;       This routine waits for the 8042 buffer to empty     :
;       Input:  None                                        :
;       Output: (AL)=0 8042 input buffer empty (ZF=1)       :
;               (AL)=2 Time out, 8042 buffer full (ZF=0)    :
;-----------------------------------------------------------:
empty_8042      PROC
        push    cx                      ;save CX
        sub     cx,cx                   ;CX=0 will be time out value
empty_8042_01:
        in      al,status_port  ;read 8042 status port
        and     al,00000010b    ;test inp. buffer full flag (D1)
        loopnz  empty_8042_01   ;loop until input buffer empty
                                ; or time out
        pop     cx              ;restore CX
        ret
empty_8042      ENDP

        SUBTTL  form_24bit_address
        PAGE
;-----------------------------------------------------------:
; FORM_24BIT_ADDRESS                                        :
;       Input:  DX has some segment                         :
;               CX has some offset                          :
;       Output: DX has base_lo_word                         :
;               CL has base_hi_byte                         :
;-----------------------------------------------------------:
form_24bit_address      PROC
        push    ax
            ;DX == s15 s14 s13 s12 s11 ... s04 s03 s02 s01 s00
        rol     dx,4
            ;DX == s11 ... s04 s03 s02 s01 s00 s15 s14 s13 s12
        mov     ax,dx
            ;AX == s11 ... s04 s03 s02 s01 s00 s15 s14 s13 s12
        and     dl,0F0h
            ;DX == s11 ... s04 s03 s02 s01 s00   0   0   0   0
        and     ax,0Fh
            ;AX ==   0 ...   0   0   0   0   0 s15 s14 s13 s12
        add     dx,cx   ;form_24bit_address
        mov     cx,ax   ;get base_hi_byte in CL
        adc     cl,ch   ;carry in (CH=0)
        pop     ax
        ret
form_24bit_address      ENDP

        SUBTTL  Real Mode re-entry point.
        PAGE
```

```
        ASSUME  ds:cseg                 ; IN REAL MODE ...
real:   sti                             ; turn the int's on
        int     20h                     ; back to DOS

        mov     ah,disable_bit20    ;gate address bit 20 on
        call    gate_a20
        mov     al,i8259_1
        out     inta01,al           ;set status of Int Controller #1
        mov     al,i8259_2
        out     intb01,al           ;set status of Int Controller #2

        sti                         ;turn the interrupts on
        int     20h                 ;back to DOS

main        ENDP
cseg_leng   EQU $
cseg        ENDS
            END  start
```

LISTING 2: METHOD2.ASM

```
        .286C   ; Tell MASM 2.0 about 286 instructions
;-------------------------------------------------------------:
;       Sample Program 2                                      :
;                                                             :
; This program switches into Protected Virtual Mode, changes  :
; the display attribute to reverse video, and returns to Real :
; Mode to exit to DOS                                         :
;                                                             :
; Once entered into a file, do the following:                 :
;       MASM SAMPLE2;                                         :
;       LINK SAMPLE2;                                         :
;       EXE2BIN SAMPLE2 SAMPLE2.COM                           :
;       DEL SAMPLE2.EXE                                       :
;                                                             :
; WARNING: This program will "kill" a PC.  It should only     :
; be run on an AT.                                            :
;-------------------------------------------------------------:
```

```
bios_data_seg   SEGMENT at      0040h
                ORG             0067h
io_rom_init     dw ?    ; dword variable in BIOS data segment
io_rom_seg      dw ?    ;  used to store a dword address
bios_data_seg   ENDS

descriptor      STRUC
seg_limit       dw      0       ; segment limit (1-65536 bytes)
base_lo_word    dw      0       ; 24 bit physical address
base_hi_byte    db      0       ;       (0 - (16M-1))
access_rights   db      0       ; access rights byte
                dw      0       ; reserved_386
descriptor      ENDS

cmos_port       equ 070h
code_seg_access equ 10011011b   ;access rights byte for code seg
data_seg_access equ 10010011b   ;access rights byte for data seg
enable_bit20    equ 11011111b   ;8042 function code to gate A20
port_a          equ 060h        ;8042 port A
shut_cmd        equ 0FEh        ;cmd to 8042: shut down AT
shut_down       equ 00Fh        ;CMOS shut down byte index
status_port     equ 064h        ;8042 status port
virtual_enable  equ 0001h       ;LSB=1: Protected Virtual Mode

        SUBTTL  Macro Definitions
        PAGE
;-------------------------------------------------------------:
; These mnemonics are not supported in MASM 2.0 therefore     :
; they are supplied here as MACROS.                           :
;-------------------------------------------------------------:
lgdt    MACRO   lgdt1
        LOCAL   lgdt2,lgdt3
        db      00Fh
lgdt2   label   byte
        mov     dx,word ptr lgdt1
lgdt3   label   byte
        org     offset lgdt2
        db      001h
```

```
        org     offset lgdt3
        ENDM
        page

lmsw    MACRO   lmsw1
        LOCAL   lmsw2,lmsw3
        db      00Fh
lmsw2   label   byte
        mov     si,ax
lmsw3   label   byte
        org     offset lmsw2
        db      001h
        org     offset lmsw3
        ENDM

jumpfar MACRO   jumpfar1,jumpfar2
        db      0EAh
        dw      (offset jumpfar1)
        dw      jumpfar2
        ENDM
        SUBTTL  Program entry point and data area
        PAGE
cseg    SEGMENT para            public          'code'
        ASSUME  cs:cseg

        ORG     100h
start:  jmp     short   main

        EVEN
gdt             LABEL   word

gdt_desc        EQU     (($-gdt)/8)*8 + 0000000000000000b
gdt1    descriptor      <gdt_leng,,,data_seg_access,>
cs_code         EQU     (($-gdt)/8)*8 + 0000000000000000b
gdt2    descriptor      <cseg_leng,,,code_seg_access,>
cs_data         EQU     (($-gdt)/8)*8 + 0000000000000000b
gdt3    descriptor      <cseg_leng,,,data_seg_access,>
ss_desc         EQU     (($-gdt)/8)*8 + 0000000000000000b
gdt4    descriptor      <0FFFFh,,,data_seg_access,>
ds_desc         equ     (($-gdt)/8)*8 + 0000000000000000b
gdt5    descriptor      <0FFFFh,,,data_seg_access,>
es_desc         equ     (($-gdt)/8)*8 + 0000000000000000b
gdt6    descriptor      <0FFFFh,,,data_seg_access,>
gdt_leng        EQU     $-gdt
                PAGE
;---------------------------------------------------------------------:
; Format of the Segment Selector Component:                           :
;                                                                     :
; +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+                   :
; |               INDEX                  +TI+ RPL +                   :
; +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+                   :
;                                                                     :
; TI = Table Indicator (0=GDT, 1=LDT)                                 :
; RPL = Requested Privelege Level (00 = highest; 11 = Lowest)         :
;---------------------------------------------------------------------:
; Format of the Global Descriptor Table                               :
;                 .------------+                +---> TI              :
;                 V            |                |++-> RPL             :
; GDT ==> +---------------+    |                |||                   :
;         |   GDT_DESC    | --+ 0000000000000000b                     :
;         +---------------+                                           :
;         |   CS_CODE     |     0000000000001000b                     :
;         +---------------+                                           :
;         |   CS_DATA     |     0000000000010000b                     :
;         +---------------+                                           :
;         |   SS_DESC     |     0000000000011000b                     :
;         +---------------+                                           :
;         |   DS_DESC     |     0000000000100000b                     :
;         +---------------+                                           :
;         |   ES_DESC     |     0000000000101000b                     :
;         +---------------+                                           :
;---------------------------------------------------------------------:

        SUBTTL  Program Main
        PAGE
;---------------------------------------------------------------------:
; MAIN                                                                :
```

```
;--------------------------------------------------------------:
        ASSUME  ds:cseg
main    PROC                            ;ES=DS=CS
        cld                             ;forward

        mov     dx,cs                   ;form 24bit address out of
        mov     cx,offset gdt           ;CS:GDT
        call    form_24bit_address
        mov     gdt1.base_lo_word,dx    ;DESC now points to gdt
        mov     gdt1.base_hi_byte,cl

        mov     dx,cs                   ;form 24bit address out of
        xor     cx,cx                   ; CS:0000
        call    form_24bit_address
        mov     gdt2.base_lo_word,dx    ;CS_CODE now points to
        mov     gdt2.base_hi_byte,cl    ; CSEG as a code segment
        mov     gdt3.base_lo_word,dx    ;CS_DATA now points to
        mov     gdt3.base_hi_byte,cl    ; CSEG as a data segment

        mov     dx,ss                   ;form 24bit address out of
        xor     cx,cx                   ;SS:0000
        call    form_24bit_address
        mov     gdt4.base_lo_word,dx    ;SS_DESC now points to
                                        ; stack segment
        mov     gdt4.base_hi_byte,cl

        lgdt    gdt                     ;Load the GDTR

        mov     ah,enable_bit20         ;gate address bit 20 on
        call    gate_a20
        or      al,al                   ;was the command accepted?
        jz      m_10                    ;go if yes
        mov     dx,offset gate_failure  ;print error msg
        mov     ah,9                    ; and terminate
        int     21h
        int     20h

gate_failure    db      "Address line A20 failed to Gate open$"

m_10:   cli                     ;No interrupts
        pushf                   ;Simulate INT by pushing Flags,
        push    cs              ; CS,
        mov     ax,offset real  ; and offset of return address.
        push    ax

        pusha           ;Now, set up stack the way BIOS block-
        push    es      ; move logic will expect it.
        push    ds      ;On return, this is the way
                        ; the regs will be.
```

```
        ASSUME  ds:bios_data_seg
        mov     dx,bios_data_seg        ;Place on Stack current SS:SP
        mov     ds,dx
        mov     io_rom_seg,ss
        mov     io_rom_init,sp

        mov     al,shut_down            ;Set shutdown byte to
        out     cmos_port,al            ; shut down code x"09"
        jmp     short   $+2             ;I/O delay
        mov     al,9
        out     cmos_port+1,al

        mov     ax,virtual_enable       ;machine status word needed
        lmsw    ax                      ; to switch to virtual mode
        jumpfar m_20,cs_code            ;Must purge prefetch queue

m_20:   ASSUME  ds:cseg                 ;IN VIRTUAL MODE ...
        mov     ax,ss_desc              ;stack segment selector
        mov     ss,ax                   ;user's ss+sp
                                        ; is not a descriptor

        mov     ax,cs_data
        mov     ds,ax                   ;DS = CSEG as data

        mov     gdt5.base_lo_word,0000h ;use 8000 for COLOR
        mov     gdt5.base_hi_byte,0Bh
        mov     gdt6.base_lo_word,0000h
        mov     gdt6.base_hi_byte,0Bh

        mov     ax,ds_desc
        mov     ds,ax
        mov     ax,es_desc
        mov     es,ax
        mov     cx,80*25
        xor     si,si
        xor     di,di
m_30:   lodsw
        mov     ah,70h                  ;attribute reverse video
        stosw
        loop    m_30

        mov     al,shut_cmd             ;shutdown cmd
        out     status_port,al          ;get back into REAL mode
m_40:   hlt
        jmp     short   m_40

        SUBTTL  Gate A20
        PAGE
```

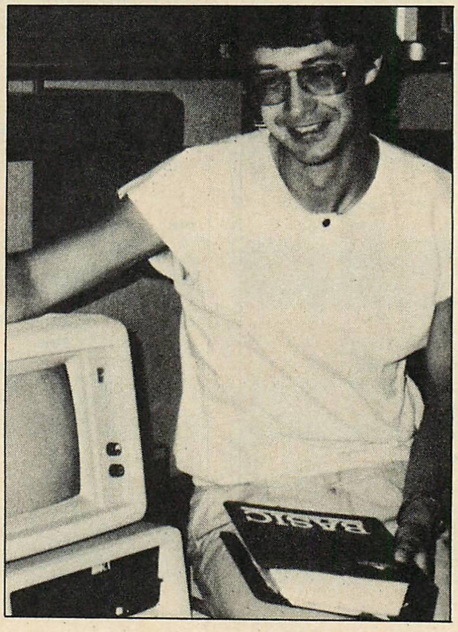
```
;-------------------------------------------------------:
; GATE_A20                                              :
; This routine controls a signal which gates address bit 20.   :
; The gate A20 signal is an output of the 8042 slave processor.:
; Address bit 20 should be gated on before entering protected  :
; mode.  It should be gated off after entering real mode from  :
; protected mode.                                       :
; Input:  (AH)=0DDh addr bit 20 gated off (A20 always 0)       :
;         (AH)=0DFh addr bit 20 gated on  (286 controls A20)   :
; Output: (AL)=0 operation successful.  8042 has accepted cmd  :
;         (AL)=2 Failure -- 8042 unable to accept command.     :
;-------------------------------------------------------:
gate_a20  PROC
        cli                      ;disable ints while using 8042
        call  empty_8042         ;insure 8042 input buffer empty
        jnz   gate_a20_01        ;ret if 8042 unable to accept cmd
        mov   al,0D1h            ;8042 command to write output port
        out   status_port,al     ;output cmd to 8042
        call  empty_8042         ;wait for 8042 to accept command
        jnz   gate_a20_01        ;ret if 8042 unable to accept cmd
        mov   al,ah              ;8042 port data
        out   port_a,al          ;output port data to 8042
        call  empty_8042         ;wait for 8042 to port data
gate_a20_01:
        ret
gate_a20  ENDP
;-------------------------------------------------------:
; EMPTY_8042                                            :
; This routine waits for the 8042 buffer to empty       :
; Input:  None                                          :
; Output: (AL)=0 8042 input buffer empty (ZF=1)         :
;         (AL)=2 Time out, 8042 buffer full (ZF=0)      :
;-------------------------------------------------------:
empty_8042  PROC
        push  cx                 ;save CX
        sub   cx,cx              ;CX=0 will be the time out value
empty_8042_01:
        in    al,status_port     ;read 8042 status port
        and   al,00000010b       ;test input buffer full flag (D1)
        loopnz empty_8042_01     ;loop until input buffer empty
                                 ; or time out
        pop   cx                 ;restore CX
        ret
empty_8042  ENDP

        SUBTTL form_24bit_address
        PAGE
```

```
;-------------------------------------------------------:
; FORM_24BIT_ADDRESS                                    :
;   Input:  DX has some segment                         :
;           CX has some offset                          :
;   Output: DX has base_lo_word                         :
;           CL has base_hi_byte                         :
;-------------------------------------------------------:
form_24bit_address  PROC
        push    ax
                ;DX == s15 s14 s13 s12 s11 ... s04 s03 s02 s01 s00
        rol     dx,4
                ;DX == s11 ... s04 s03 s02 s01 s00 s15 s14 s13 s12
        mov     ax,dx
                ;AX == s11 ... s04 s03 s02 s01 s00 s15 s14 s13 s12
        and     dl,0F0h
                ;DX == s11 ... s04 s03 s02 s01 s00   0   0   0   0
        and     ax,0Fh
                ;AX ==   0 ...   0   0   0   0   0 s15 s14 s13 s12
        add     dx,cx    ;form_24bit_address
        mov     cx,ax    ;get base_hi_byte in CL
        adc     cl,ch    ;carry in (CH=0)
        pop     ax
        ret
form_24bit_address ENDP

        SUBTTL      Real Mode re-entry point.
        PAGE
        ASSUME ds:cseg              ;IN REAL MODE ...

real:   mov     dx,cs
        mov     ds,dx               ;DS = CS
main            ENDP

cseg_leng       EQU     $
cseg            ENDS
                END     start
```

"I Program In BetterBASIC And I Recommend It."

"My company uses BetterBASIC to control the automation system on board ships. It gives us the excellent feature of flexibility and quick change of process dependant logic. We save the logic in E²prom (electrically erasable programmable ROM) so online changes can be made instantly. I use BetterBASIC and I recommend it."

Ken Moller · Norcontrol · Horten, Norway



640K Now you can use the full memory of your PC to develop large programs. **STRUCTURED** Create well organized programs using procedures and functions that are easily identified and understood and completely reusable in future programs. **MODULAR** Use procedures and functions grouped together to form "library modules" which are then available to you or anyone else for future use. **EXTENSIBLE** Create your own BetterBASIC modules which contain BetterBASIC extensions. This feature coupled with the easy-to-use Assembly Language support, makes this an ideal OEM language. **INTERACTIVE** BetterBASIC acts like an interpreter because it responds to the users' commands in an immediate mode. However each statement is actually compiled as it is entered. **COMPILED** Each line of the program is compiled as it is entered into the computer's memory rather than interpreted at runtime. **RUNTIME SYSTEM** The optional Runtime System generates stand alone EXE. files allowing for the distribution of products written in BetterBASIC with no royalties. **SUPPORTS** Windows, Graphics, DOS and BIOS ROM calls, Chaining, Overlays, Local and Global Variables, Recursion . . . and more. **BetterBASIC** Runs on IBM PC, XT, AT and all IBM-compatibles. Ask your local dealer for BetterBASIC or call 1-800-225-5800 in Canada call 416-469-5244. Also available for the Tandy 1000, 1200, AND 2000 AT Tandy/Radio Shack stores.

Summit Software Technology, Inc.™
P.O. Box 99, Babson Park
Wellesley, MA 02157

PRICES:

BetterBASIC	$199
8087/80287 Math Module	$99
BTrieve™ Interface	$99
Runtime System	$250
Sample Disk with Tutorial	$10

Better BASIC.

Because It's The Best.

CIRCLE NO. 179 ON READER SERVICE CARD

MasterCard, Visa, Checks, Money Order, C.O.D. accepted and P.O. on approval.

(If you're using BetterBASIC and would like to be featured in one of our ads, please write to the Director of Advertising at Summit.)

Boilerplate Deal Killers

Form contracts, intended to protect the vendor, may backfire and scare the customer away.

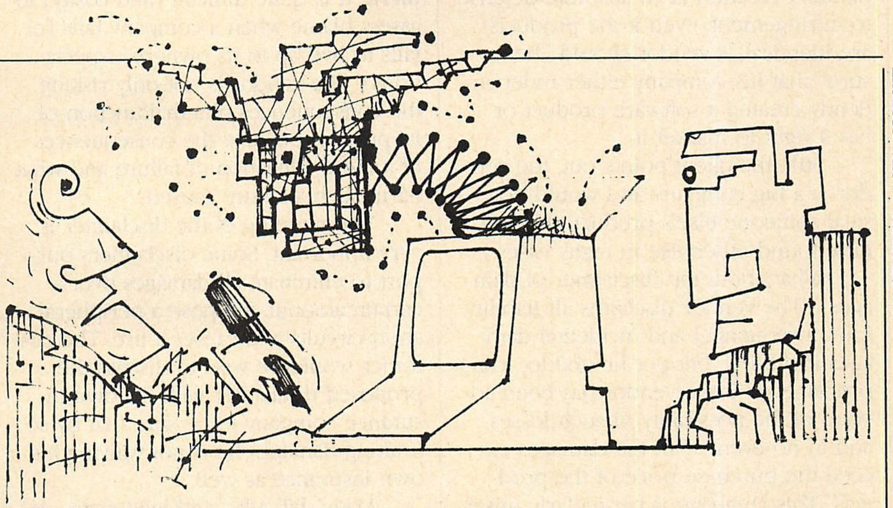


ILLUSTRATION • MACIEK ALBRECHT

Buying computer hardware and software is more complicated than buying the traditional business staples. The harm that can result from the wrong decision is much greater, and most customers do not have the skill or time required to evaluate computers as they would a typewriter.

Many large companies have specifications and testing procedures of varying degrees of sophistication to determine whether a computer product has the required features; most smaller companies rely on the salesperson to tell them what the product will do. The choice of systems often comes down to the salesperson's persuasion and the vendor's reputation. Often, the decision is made because of a "good feeling" about the vendor and its people.

Too often, the last step in the process of purchasing computer products is the vendor's presentation of its printed contract. The customer would do better to examine this form before looking at the product. Many deals fall through after months of analysis of the relative merits of competing systems, because the winning vendor presents a form contract that undoes all of the confidence that the salesperson has worked so hard to build.

Economics usually dictates that a vendor of high-volume goods or services present a form contract rather than negotiate each transaction from scratch. Generally, a form contract will contain certain provisions intended to be modified to fit the particular deal, and other *boilerplate* provisions that are meant to stand as written. But good financial sense also dictates that the boilerplate not be so outrageous that it forces the customer to go elsewhere (or make the negotiation so protracted as to eliminate the cost savings realized from having a form contract).

Some sample boilerplate provisions follow. These are purely hypothetical, but each element is commonplace.

A salesperson has spent months evaluating a customer's operations and explaining how his system will meet the customer's needs better than competing systems. The customer agrees, and then is presented with a contract that says, "Vendor disclaims all warranties including the warranties of merchantability and fitness for purpose."

The customer calls his lawyer for an explanation of this provision. It means (although whether it is enforceable or not is another question altogether) that the vendor is not guaranteeing that his system will do anything in particular. "What," asks the customer, "about all the promises the salesperson made? I have several brochures."

Very pretty, but look at the contract section that says, "This contract contains the complete agreement between the parties and supercedes any prior agreements, whether written or oral."

A lawyer is not needed to interpret that language. (I generally suggest modifying the language to say "This contract and attachments. . . " and then attaching all the brochures and sales literature. Some vendors resist even that. One told me, "We can't do that—those are only sales brochures.")

The client now says that the salesperson told him not to worry about that paragraph—it is only something the vendor's lawyers make them put in the contract. This explanation is hardly comforting, given the contract provision:

> This contract may not be modified orally and may only be modified on behalf of the vendor by a written document signed by a duly authorized officer of vendor at its home office.

Did the salesperson say that he was an officer of the corporation? Even if he is an officer, how do we know he has been "duly authorized?" Furthermore, he must put his modification in writing if it is to be valid.

Undaunted, the customer points out that he has seen a demonstration of the system and is confident that it will work—can we please go on to the important aspects of the contract?

How about the paragraph that says the customer agrees to indemnify and hold harmless the vendor against any claims of infringement? The vendor claims to own the product he is selling, yet he is asking that the customer pay the vendor's costs if someone sues the vendor claiming that the customer's use of the product infringes a copyright or patent. This could be compared to a customer buying a new car for $12,000 with the understanding that if the dealer had stolen the car, the customer would pay both the dealer's legal fees and the $12,000 cost of the car! If any-

thing, the vendor should be indemnifying the customer.

Software vendors have argued that they should not indemnify the customer because, although they wrote the program themselves and they own it, they "cannot be sure and their lawyers will not let them make any warranties." Such an argument may have some validity in the case of patents (where independent invention is not a defense to infringement), but with copyrights independent creation is an absolute defense to infringement, even if the products are identical. A vendor should "be sure" that his company either independently created a software product or has a right to market it.

Still, the client points out, the vendor is a big company and would not steal someone else's product. (Courts have found otherwise in many cases.)

What about the disclaimer of damages? "The vendor disclaims all liability for consequential and incidental damages including, but not limited to, lost profits, even if the vendor has been advised of the possibility of such losses, and in no event shall the damages exceed the purchase price of the product." This language is particularly unsettling if the customer has spent six months explaining his company's operations to the vendor, who now knows the customer is gambling his whole company on the product's continuing performance. Often the vendor argues, "This is a $10,000 piece of software. If we don't sell it to you, our company is out only $10,000—not even that, because we'll sell it to someone else. It's unreasonable to ask us to risk millions just to make this sale."

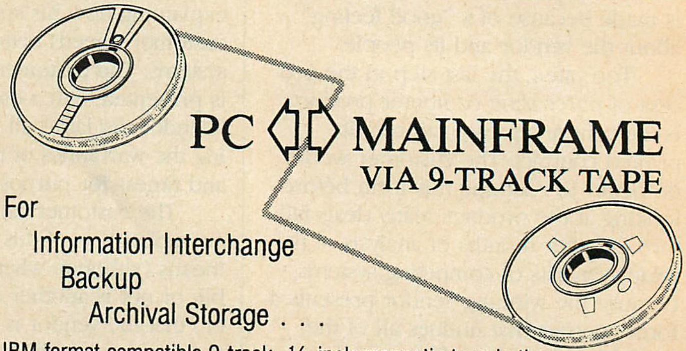
In all fairness, this argument has merit. It is quite difficult (and costly) to assess blame when a company fails (or fails to live up to its owners' expectations). The vendor is not only risking the consequences of a malfunction of its product but also the consequences of a mere allegation of failure and what harm such a failure caused.

The wording of the disclaimer is very important. Some disclaimers purport to eliminate all damages over a certain amount. Suppose a peripheral short-circuits and causes a fire. The customer would be wise to discuss the proposed disclaimer with his own insurance company to be sure that by so limiting damages he is not waiving his own insurance as well.

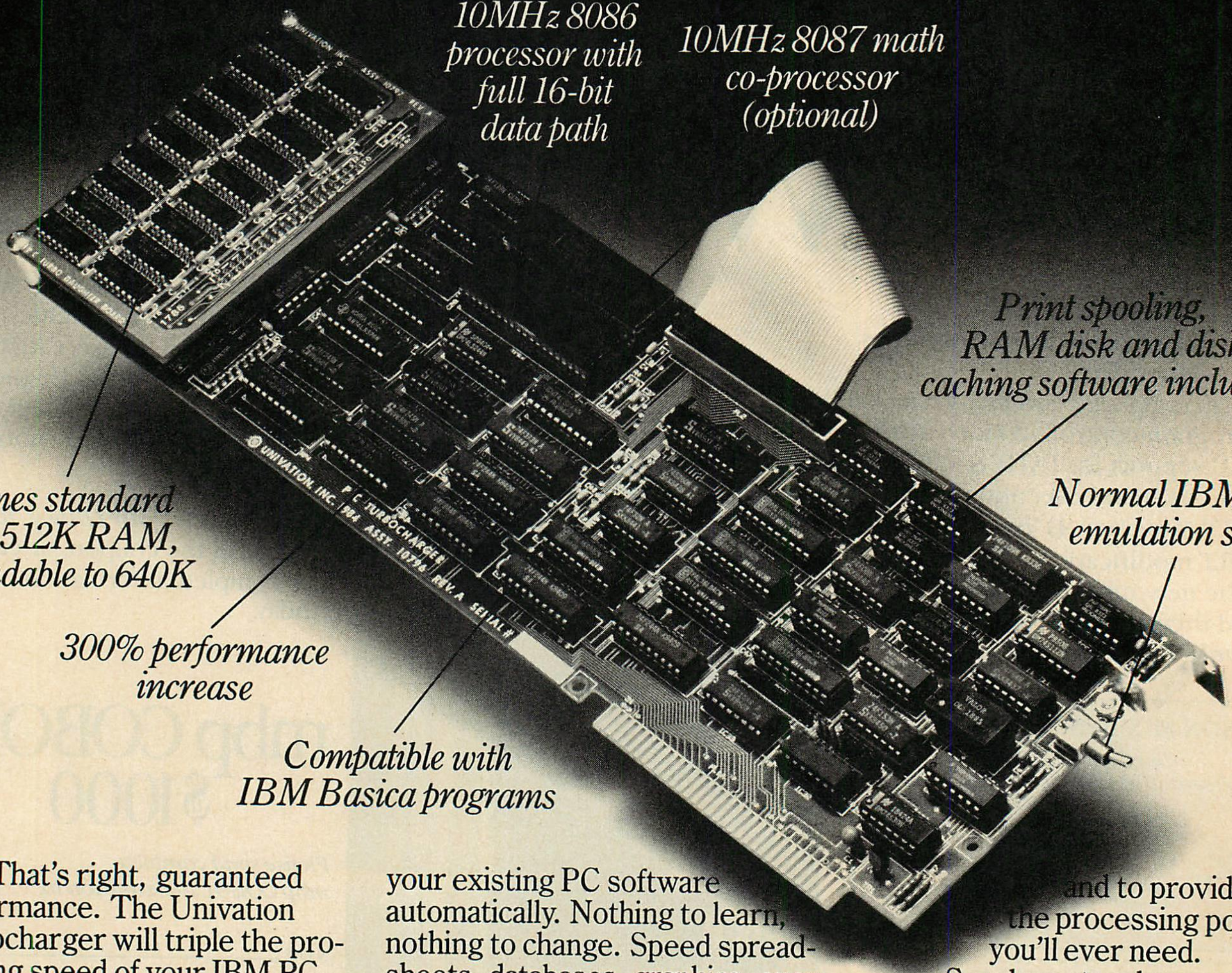
Many difficult—and legitimate—issues must be considered in negotiating computer hardware and software acquisitions. The vendor's boilerplate provisions discussed above are difficult to defend if the customer challenges them. A customer may be so exhausted by the evaluation of the system that he accepts the vendor's form contract without analysis. If the customer has any stamina remaining, however, such boilerplate provisions can undo much of the work the salesperson has done to get the customer to this point.

It is therefore important not only from the customer's viewpoint, but also from the vendor's, to give careful thought to which party should accept which risks. Some vendors have form contracts that state, "We are going to stand behind our product—we only await protection from the customer's mistakes." In their view, the risk is small because they have confidence in themselves, the risk is appropriately theirs, and they hope that the benefit is substantial because increasingly sophisticated customers will gravitate toward vendors who are prepared to stand behind their products.

Max Stul Oppenheimer, PC, is a partner in the law firm of Venable, Baetjer & Howard, located in Baltimore.



PC AT PERFORMANCE FOR A FRACTION OF THE COST. GUARANTEED.



That's right, guaranteed performance. The Univation Turbocharger will triple the processing speed of your IBM PC, PC/XT, or 100% compatible.

Just think, the data processing speed and performance of a PC/AT without the expense. Plug it in, and your PC will do everything it did before, only 300% faster.

Best of all, the Univation Turbocharger works with all your existing PC software automatically. Nothing to learn, nothing to change. Speed spreadsheets, databases, graphics, even IBM Basica programs. The Turbocharger will spark each and every program you use.

To ensure complete reliability, our Turbocharger is extensively tested. It's available for immediate delivery. And, best of all, it's guaranteed. Guaranteed to work exactly as promised and to provide all the processing power you'll ever need.

So why not make your life a little easier and do your work a lot quicker with the Univation Turbocharger.

See it at your local computer dealer today. For the dealer in your area call: (408) 745-0180.

CIRCLE NO. 202 ON READER SERVICE CARD

UNIVATION
YOUR EXPANSION COMPANY

1037 North Fair Oaks Ave.
Sunnyvale, CA 94089
(408) 745-0180

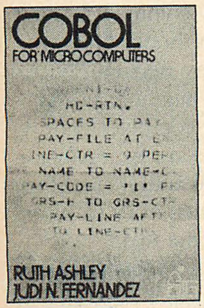
Dealer and distributor inquires invited.

IBM PC, PC/XT and PC/AT are trademarks of International Business Machines Corp.

Learning COBOL

Microcomputer users unfamiliar with COBOL need a little help understanding the language.

COBOL for Microcomputers, A Self-teaching Guide
Ruth Ashley and Judi N. Fernandez
(John Wiley & Sons, New York 1983)
326 pages, paper, $15.95



Compiler documentation rarely includes any tutorial material for teaching new users how to program in the particular language. This is especially true of COBOL compilers, whose primary market is expected to be among programmers who have been trained on either mainframes or minicomputers. *COBOL for Microcomputers, A Self-teaching Guide* has been written for the microcomputer user who has little or no COBOL experience, but who has a need or desire to learn this programming language.

The book is not meant as an introduction to the principles of programming or compilers in general, but then COBOL is not usually the first compiled language that a microcomputer user learns. Nevertheless, the first chapter presents a brief review of general programming principles, with a concise introduction to structured programming concepts. Depending on how it is written, COBOL can be as highly structured as Pascal or as spaghetti-coded as BASIC, so one of the major goals of any COBOL tutorial should be to impress the necessity for structuring on the new user. In this regard, *COBOL for Microcomputers* is fairly successful.

The organization of the book reflects its intended use as a self-teaching text. Material is presented in small amounts, frequently with program or pseudocode examples; the reader is asked a series of questions or directed to modify or expand each example. The answers are given immediately following each set of questions, usually on the same or facing page. This scheme should be equally acceptable to those readers who want to try their hand at implementing the principles they just read about, as well as those who prefer to learn just by reading explanations and ready-made examples.

In keeping with the structured programming approach, problems are presented not with flowcharts, but with hierarchical structure diagrams. For someone who is totally new to programming, these diagrams perhaps are not as intuitive as flowcharts, but those readers with some programming experience will find that coding from structure diagrams usually results in more highly structured programs.

Although the book is quite useful for learning the capabilities of COBOL, it is less successful in teaching how to operate a COBOL compiler on an IBM PC. Three compilers are covered: Microsoft COBOL, Ryan-McFarland (RMCOBOL), and Nevada COBOL, but the instructions provided are for a CP/M operating system, not for PC-DOS. Nevada COBOL is not even available for the IBM PC, and the DOS interface of Microsoft's current COBOL compiler is quite different from the one described in the book. The implementation of RMCOBOL, and the differences between it and the Microsoft COBOL language, were essentially unchanged when these compilers were ported from CP/M to MS-DOS; therefore, the sections dealing with RMCOBOL and its differences from Microsoft still are relevant.

All three of these compilers, in their CP/M incarnations, were fairly low-level implementations of COBOL; accordingly, the book does not cover some of the more useful advanced features, such as alternate indexes, file sorting, and string manipulations.

The standard features of the language are covered in a fair amount of detail. Console I/O is presented first, because that is a prerequisite for useful programs. Extended forms of the ACCEPT and DISPLAY statements for the Microsoft and Ryan-McFarland COBOL compilers are explained, but Microsoft's full-screen mode is not. Subsequently, data structures and program control structures are introduced. The book's most advanced chapters deal with file I/O, covering operations with multiple sequential files and with indexed files in sequential, random, and dynamic access modes. Relative files, however, are not mentioned.

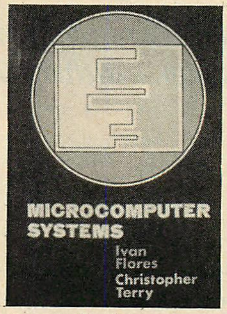
An anomaly in this age of interactive computing is the authors' suggestion that programs should be initially written on a coding form. Most microcomputer users have not even seen one of these. Because COBOL source programs require a fixed format (meaning that statements have to be spaced to exact specifications on each line), a coding form might be a useful device for introducing and explaining this format. Most users, however, will probably prefer to enter code directly into a full-screen editor.

Just as word processors have revolutionized writing, so program editors have changed the generation of source code, in any programming language, from a paper-and-pencil exercise to an interactively creative process. Despite the fairly recent publication date of this book, the authors seem to be several years behind the times in terms of operating systems, compiler capabilities, and programming practices.

Overall, the book is easy to read and could be useful for the new user of either Microsoft COBOL or RMCOBOL. It needs to be supported by strong compiler documentation, such as that provided by both Microsoft and Ryan-McFarland. It cannot be recommended, however, for novice users who are not generally familiar with compiled languages and compiler operations.

—TED MIRECKI

Microcomputer Systems
Ivan Flores and Christopher Terry (Van Nostrand Reinhold Co., New York, NY, 1982) 290 pages; cloth, $24.50



This could be it! *Microcomputer Systems* could be the book that ties together all of those fragmented facts and factions that make up small computer systems. Users frequently fail to grasp the bigger picture of a complete operating system. They work with an individual language or specific hardware and do not have an appreciation of how all the pieces fit together.

Flores and Terry comment in their preface that computer books fit into two general categories: beginner books and technical books. Beginner books deal with elementary computer concepts; technical books are generally for design engineers. The authors feel they have written a book that finds its place somewhere between the two.

In the past two years this text has been used as the principal reference in a sophomore course on small systems at Broome Community College in Binghamton, New York. The course is the last in a sequence of hardware courses that includes computer logic and an introduction to microprocessors. In addition to other computer courses, students receive mainframe experience on a DEC computer and an advanced Pascal course on an IBM PC. The small-systems course is designed to tie these hardware and software learning experiences together. *Microcomputer Systems* has helped educators achieve that goal.

The book is divided into 16 chapters, plus a glossary and an index. The chapters are grouped into the larger categories of introductory material, entry and presentation, storage, and system integration. The introduction (chapters 1 and 2) gives the reader a quick overview of hardware and software concepts. Chapter 1 discusses the microprocessor, memory, interconnections, buses, interfaces, and I/O addressing. Chapter 2 covers systems software, bootstrapping and loading, programming aids, language processors, and translators. Many of these topics are covered in detail in later chapters.

The Entry and Presentation section of the book is presented in chapters 3 through 8, which cover in detail the

Tools That Make Your Job Easier

For PCDOS/MSDOS (2.0 and above/128K) • IBM PC/Compatibles, PC Jr., Tandy 1000/1200/2000, & others
For CPM80 2.2/3.0 (Z80 required/64K) • 8" SSSD, Kaypro 2/4, Osborne I SD/DD, Apple II, & others

MIX EDITOR

Programmable, Full/Split Screen Text Processor

Introductory Offer **29.95**

Great For All Languages

A general purpose text processor, the MIX Editor is packed with features that make it useful with any language. It has auto indent for structured languages like Pascal or C. It has automatic line numbering for BASIC (255 character lines). It even has fill and justify for English.

Split Screen

You can split the screen horizontally or vertically and edit two files simultaneously.

Custom Key Layouts

Commands are mapped to keys just like WordStar. If you don't like the WordStar layout, it's easy to change it. Any key can be mapped to any command. You can also define a key to generate a string of characters, great for entering keywords.

Macro Commands

The MIX Editor allows a sequence of commands to be executed with a single keystroke. You can define a complete editing operation and perform it at the touch of a key.

Custom Setup Files

Custom keyboard layouts and macro commands can be saved in setup files. You can create a different setup file for each language you use.

MSDOS Features

Execute any DOS command or run another program from inside the editor. You can even enter DOS and then return to the editor by typing exit.

MIX C COMPILER

Full K&R Standard C Language
Unix Compatible Function Library

Introductory Offer **39.95**

Complete & Standard

MIX C is a complete and standard implementation of C as defined by Kernighan and Ritchie. Coupled with a Unix compatible function library, it greatly enhances your ability to write portable programs.

The Best C Manual

MIX C is complemented by a 400 page manual that includes a tutorial. It explains all the various features of the C language. You may find it more helpful than many of the books written about C.

Fast Development

MIX C includes a fast single pass compiler and an equally fast linker. Both are executed with a simple one line command. Together they make program development a quick and easy process.

Fast Execution

The programs developed with MIX C are fast. For example, the often quoted prime number benchmark executes in a very respectable 17 seconds on a standard IBM PC.

Standard Functions

In addition to the functions described by K&R, MIX C includes the more exotic functions like *setjmp* and *longjmp*. Source code is also included.

Special Functions

MIX C provides access to your machine's specific features through BDOS and BIOS functions. The CHAIN function lets you chain from one program to another. The MSDOS version even has one function that executes any DOS command string while another executes programs and returns.

Language Features

- Data Types: char, short, int, unsigned, long, float, double (MSDOS version performs BCD arithmetic on float and double-no roundoff errors)
- Data Classes: auto, static, extern, register
- Struct, Union, Bit Fields (struct assignment supported)
- Typedef, Initialization
- All operators and macro commands are supported

30 DAY MONEY BACK GUARANTEE

Orders Only: **Call Toll Free 1-800-523-9520,** (Texas only **1-800-622-4070**)

MIX Editor ____ $29.95 + shipping ($5 USA/$10 Foreign) Texas residents add 6% sales tax
MIX C ____ $39.95 + shipping ($5 USA/$25 Foreign) Texas residents add 6% sales tax

Visa ____ MasterCard ____ Card # ____ Exp. Date ____
COD ____ Check ____ Money Order ____ Disk Format ____
Computer ____ Operating System: MSDOS ____ PCDOS ____ CPM80 ____
Name ____
Street ____
City/State/Zip ____
Country ____
Phone ____

MIX software
2116 E. Arapaho
Suite 363
Richardson, Tx 75081
Dealer Inquiries Welcome
Call (214) 783-6001

T

common input and output options of a small system. The chapter on keyboards details how various keyboards are constructed and decoded. The video display chapter covers such topics as raster generation, refresh memory, character generation (with block diagram circuits), screen geometry, screen timing, row generation, frequency references, screen updating, and cursor control.

Chapter 5, a fairly short chapter without excessive detail, explains the differences between dumb, smart, and intelligent terminals. In the next two chapters, the authors discuss all types of printers from the original Teletype models 33 and 35 to ink jet printers (laser printers are not mentioned). This discussion includes a section on speed versus quality, daisy wheels and thimbles, platen and carriage mechanisms, bands and belts, dot matrices, bidirectional printing, multiple print sizes, and multiple-pass printing.

The final chapter of this section discusses various options available for displaying graphics. The treatment of refresh rasters, stroke-written displays, stroke-written storage-tube displays, hard copy printers and plotters, X-Y coordinate digitizers, and light pens amounts to not much more than a definition of the device and a short description. Certainly, this chapter could and should have been expanded.

Chapter 9 introduces the reader to external storage devices: punched cards, paper tape, and magnetic tape; this brief chapter discusses tape geometry and coding methods. Chapter 10 provides the real insight into magnetic recording methods; it is divided into two major categories: analog and saturation (digital) recording techniques. The section dealing with analog recording techniques discusses single-tone on-off, single-tone pulse ratio, frequency-shift keying (FSK), and dual-tone phase encoding. Also included is a fairly detailed discussion of wave-shaping as it relates to analog recording techniques. In the section detailing digital recording methods, the authors cover frequency modulation (FM), modified frequency modulation (MFM), modified modified frequency modulation (MMFM), and group-coded recording (GCR).

Hardware chapters 11 and 12 cover cassette, cartridge, and disk drives. In addition to a discussion of the hardware involved, the authors provide a detailed account of the "logic" required to record and play back information. A discussion of disk controllers and hard disks is also included.

Another deluxe chapter (14) deals with telecommunications. The authors go beyond a typical discussion of modems and include techniques for short, intermediate, and long communications links. The detailed discussion of half-duplex, full-duplex, and echoplex methods centers around 300 to 1200 bits per second (bps) transmission.

The last two chapters are supposed to cover system integration, but they are disappointing. In chapter 15 (Operation of a Typical Operating System) the authors discuss CP/M, although not very completely, as a typical operating system, but then they do not include any example small systems that use CP/M in chapter 16 (Some Real Systems). The result is a lack of chapter integration *and* system integration.

Overall, however, I can recommend *Microcomputer Systems* as a satisfactory tool for giving a clear shape to the total small-systems picture; it falls short in only a few areas.

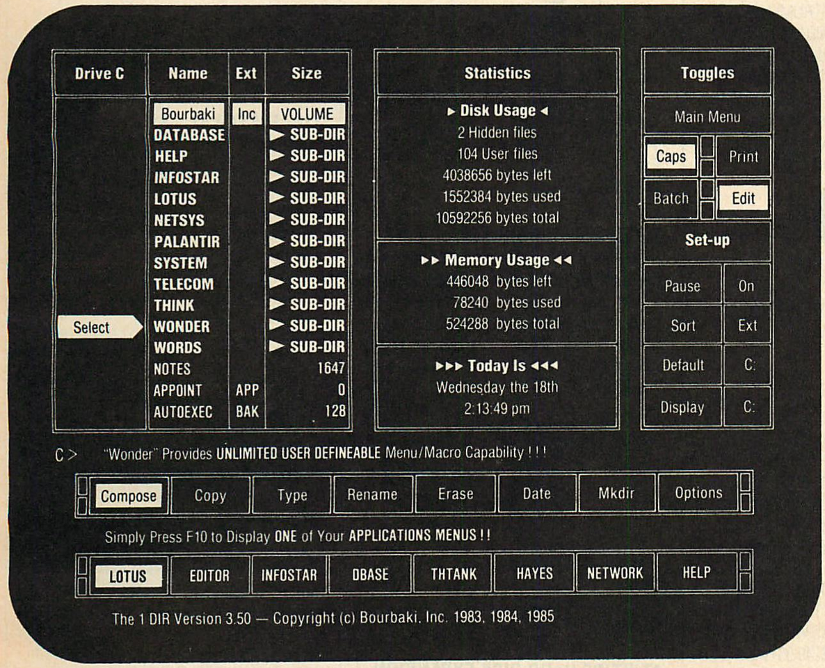
—WILLIAM H. MURRAY

Which Would You Prefer?

A>_

This
or
This

Customize Your System
with the Ultimate
"Productility"



the 1 dir
"Wonder"

EASY to USE "VISUAL SHELL" for DOS

"WONDER" integrates a powerful File Management System with extensive user definable Menu/Macro capability. Create menus and program single keystroke commands to run any or all of your applications. Create your own customized HELP system for yourself or other users. In addition, version 3.5 provides full color support, menu password protection, CUSTOM multiple file operations, CUSTOM prompted commands, plus much more.

If you are a member of the growing population of new PC users, whose interest lies primarily in taking advantage of the computer as information management tool, this program was designed for you. If you have a hard disk, you shouldn't be without it! ***1* dir** is licensed by more hard disk manufacturers than any other visual shell.

MICRO D Corporate Training Specialist — Mr. Don Kenny

"What impresses me most about ***1* dir** is how much you have intelligently added to it over the last few months. It is very rare to find a company such as yours with a practical knowledge of how computers can be tamed . . . ***1* dir** is still the **BEST** operating environment I've ever seen on a computer."

"WONDER" is also a *perfect* Training Tool that grows *with* the *user!*
A real *problem solver* for MIS Managers and Systems Integrators.

Works with the IBM PC/XT/AT family. (DOS 2.00 — or higher), plus compatibles.

Call, write or see your local dealer for more information!
Evaluation copies available to dealers and corporate clients.

by Bourbaki Inc.™

$95.00

P.O. Box 2867, Boise, ID 83701, (208) 342-5849

CIRCLE NO. 235 ON READER SERVICE CARD

TECH BOOK

ACCESSORIES/SUPPLIES

BINDERS, SLIPS, DISKS
Complete program packaging line. IBM style D-ring binders, slip cases, floppy pages, game portfolios. Continuous paper with three large holes, 20 lb. to go in binders. Blank & printed disk envelopes. Function key cards for your F1-F10 keys. Bulk disks. Disk labels. Call for FREE catalog & prices. Fast service, low $.
ANTHROPOMORPHIC SYSTEMS LTD.
376 E. St. Charles Road
Lombard, IL 60148
(312) 629-5160

BUS. OPPORTUNITIES

Personal Computer Owners
CAN EARN $1,000 TO $5,000
monthly selling simple services performed by their computer. Work at home-in spare time. Get free list of 100 best services to offer. Write:
C.I.L.B.O.
PO Box 60369
San Diego, CA 92106-8369

EXTRA INCOME
.. Learn 3 steps to Profitable Consulting
.... How to Locate Prospects
.... How to Bid on Projects
.... How to Reference Sell New Clients
.. Discover 4 Profitable Client Services
Invest $9.95 in "Make Your PC Profitable."
FREE Table of Contents sent on request.
TULSA COMPUTER CONSORTIUM (TCC)
P.O. Box 707
Owasso, OK 74055
918-747-0151

CONSULTANTS

PC SOFTWARE DEVELOPMENT
Save time and money by having our professional engineering team develop or port your software. We will code to your specifications and test criteria. Development in C, Pascal or Assembly for the IBM PC running PC-DOS, Unix or Zenix. Results are guaranteed, rates are flexible and the finished product belongs to you!
UNISULT, INC.
One Devonshire Place, Suite 1409
Boston, Massachusetts 02109
(617) 742-0374

HARDWARE/ADD-ON BOARDS

LOW COST MODULAR DAS SYSTEM
Now available Modular INSTRUMENTAL DATA ACQUISITION SYSTEM (MIDAS) for IBM PC family. Hardware provides direct connection to standard transducers. SIGNAL CONDITIONING INCLUDED. Software provides PC DOS drivers for compatibility with other application programs. Features include 10w unit cost, high accuracy, modularity, and expandibility.
ADVANCED PERIPHERALS INCORPORATED
12650 West Geauga Plaza
Chesterland, OH 44026
(216) 729-3927

"MULTIFONT"
A NEW ADD-ON CARD FOR THE IBM PC/XT. "MULTIFONT" ENABLES THE IBM MONOCHROME PRINTER ADAPTER TO HAVE 256 USER DEFINABLE SYMBOLS. IT COMES WITH A POWERFUL CHARACTER/SYMBOL EDITOR, A PRINTING UTILITY, AND TEN ALREADY-MADE FONTS, INCLUDING APL, HEBREW, ARABIC, SCIENTIFIC, AND ELECTRONIC SYMBOLS. "MULTIFONT" IS TOTALLY TRANSPARENT TO THE SOFTWARE AND WORKS WITH ANY WORD PROCESSORS. LIST PRICE $249.95 HASH TECH CO., 2065 MARTIN AVE. #103, SANTA CLARA, CA 95050, (408) 988-2646.

DT2801 SERIES ANALOG I/O
Plug-in data acquisition boards with 8DI/16SE analog inputs, high or low level programmable gains, 2 analog outputs, 16 lines of digital I/O, DMA, on-board clock, and on-board microprocessor. Optional software subroutines and screw terminal panels.
DATA TRANSLATION INC.
100 Locke Drive
Marlboro, MA 01752
(617) 481-3700

IEEE-488 INTERFACE
Halfsize IEEE-488 interface and software turns your PC, AT or compatible into an instrumentation work station for the lab, test, and measurement. We are the major OEM supplier with interfaces for 10 different architectures and the most comprehensive coverage of operating systems and languages. $385 software $75.
NATIONAL INSTRUMENT
12109 Technology Blvd: Dept. PC
Austin, Texas 78727
(800) 531-5066, (512) 250-9119

$200 PER MEGABYTE
Memory expansion for your IBM PC, XT, OR AT at $200 per megabyte. All the memory you've dreamed of at a price you can live with. JRAM-2 with JRAM-AT hold up to 2 megabytes per board with Ser/Par/Clks also available. Megamemory and I/O ports all in one slot!
Tall Tree Systems
1120 San Antonio Road
Palo Alto, CA 94303
(415) 964-1980

FIXED DISK BIOS/BOOT
FiXT boots from most popular Hard Disks—DAVONG, XEBEC, IOMEGA, GT LAKES, etc. Adds XT-like BIOS interface to your disk for PC. Security, multiple volumes, removable media support optional. No-slot plug-in installation. Specify controller and computer with order. $80-$180. Add $3 shpg, CA tax.
GOLDEN BOW SYSTEMS
P.O. BOX 3039
SAN DIEGO, CA 92103
(619) 298-9349

AT/XT/PC HARD DISK EXPANSION
Replace hard disk with a bigger one, or add a second drive! Vfeature supports high capacity drives on standard AT, XT, and compatible hard disk controllers. Includes multiple volumes, security features, selectable clusters, keyboard lock. $70-$120 + $3 shipping + CA tax.
GOLDEN BOW SYSTEMS
P.O. BOX 3039
SAN DIEGO, CA 92103
(619) 298-9349

HARDWARE/PERIPHERALS

A/D TO RS232 CONVERTER
Low cost stand-alone data acquisition board for any 8 bit RS 232 port. 8 channels, 8 bit conversion. 0-5V input, 300-9600 baud. Small size, 4" × 5", with wall plug power supply. Easy-to-use, includes sample driven software listings. Kit w/all parts & PS, $59.95 + $4 shipping. Assembled & tested is $89.95 + $4.
TPS SYSTEMS
14820 Elmore Road
Anchorage, AK 99516
(907) 345-6730

BARCODE READERS
- IBM PC/XT, AT keyboard or RS-232 interface—
- NO programming required for IBM PC/XT, AT keyboard interface unit.
- Reads dot-matrix & printed labels accurately
- Choice of digital wands
- Reads UPC A/E, code 39, codabar & I 2 of 5
- $595/$635 list.

PERCON
2190 West 11
Eugene, OR 97402
503-344-1189

33% FASTER AT—$19.95
If you have a PC AT you can plug in a new crystal, and watch it run 1/3 faster. Non-technical 10 minute installation (no soldering), and no expansion slot usage. Compatible with all software. Satisfaction guaranteed. Check or Money order. COD. $3 extra.
AVTEK ENGINEERING
P.O. Box 21-171
Eagan, MN 55121
(612) 452-7415

HARDWARE COMMUNICATION

THE VISIONARY 1200
The Visionary Electronics super-smart modem sends and receives messages anytime, day or night, with or without a computer. Stores files, names, phone numbers, and more in RAM; complete communications software in ROM; on-board realtime clock; 8085 driven; NiCad memory back-up; and TELEX terminal emulation.
Visionary Electronics Incorporated
VISIONARY ELECTRONICS
141 Parker Ave.
San Francisco, CA 94118
(415) 751-8811

HARDWARE/DISK DRIVES

1.2MB PC-AT DISK DRIVES
AT your PC by replacing your diskette controller and adding a high-density disk drive. JDISKETTE handles 1 to 4 diskette drives plus and I/O module. $249 for JDISKETTE-SA, $250 for TEAC 55GF (1.2MB); $200 for TEAC 55B (36-K); $180 for Ser/Par/Clk module.
TALL TREE SYSTEMS
1120 San Antonio Rd
Palo Alto, CA 94303
(415) 964-1980

RATES AND INFORMATION

Standard listings consist of a bold lead line (23 characters maximum); 7 lines of body copy (45 characters per line); 4 lines for company name, address and telephone number. $110 per insertion—3 issue minimum. Additional charge for extra lines and company logos. Prepayment and frequency discounts available. American Express, MasterCard, Visa accepted. Copy subject to publishers approval. Send typewritten or printed copy, reproducible logo art (if applicable) and remittance to Kathryn Cumberlander, Classified Sales Manager, Ziff-Davis Publishing Company, One Park Avenue, New York, NY 10016. For additional information, assistance, or to place an order by phone, call collect (212) 503-5115.

TECH BOOK

PUBLICATIONS/ CATALOGS

COMPUTER ERGONOMICS
Your source book on user friendly design. 61 basic works reviewed and abstracted to save you time and money. Includes free catalog. Send for Computers: The Non-Technological (Human) Factors. $34.95. ALSO AVAILABLE in this series Artificial Intelligence volumes 1 and 2. $295 for the set.
REPORT STORE PUBLISHING
910 Massachusetts, Suite 503, Dept. PC
Lawrence, KS 66044
(913) 842-7348

SOFTWARE/ARTIFICIAL INTELLIGENCE

A.D.A. A.I. PROLOG-LISP TOOLKIT
A.I. Toolkit. A.D.A. PROLOG and Cybermetrics. UNX LISP in one package. Virtual memory PROLOG has full C&M syntax. LISP has macros, catch, throw, optional and rest parameters, structure editor and utilities w/source. Both systems have tree structured object list. Use with any text editor. $149.95
A.D.A.
1570 Arran Way
Dresher, PA 19025
215-646-4894

SOFTWARE/BUSINESS OPPORTUNITIES

RETRACTION
Below is the CORRECTED version of an advertisement which appeared in our May June July issues

SOFTWARE AUTHORS
WE WILL CONNECT YOU TO MAJOR SOFTWARE PUBLISHERS FREE. We are in contact with many publishers looking for marketable software. If you have a quality software package or program and wish to have it published and distributed, contact:
SOFTSEARCH, Inc.
P.O. Box 281
Budd Lake, NJ 07828
(201) 627-1790

Attention:
SOFTWARE DEVELOPERS
We are interested in receiving your completed package and business plan. Hammer Computer Systems, Inc. is in the software publishing business. Please send inquiries and complete portfolio to: Thomas A. Horvath, Vice President of Sales and Marketing.

HAMMER

HAMMER COMPUTER SYSTEMS, INC.
700 Larkspur Landing Circle/Suite 285
Larkspur, CA 94939
(415) 461-7633

VERTICAL MARKET OPPORTUNITY
If you develop software for the Advertising, Publishing or Broadcast industries call us. We're Interactive Market Systems. We have 10,000 clients in those industries World-Wide. If your product can benefit our client base, we'll provide marketing and sales support. If you currently have a product or are about to develop one give us a call. Contact Tami Peterson.
IMS Micros
355 Lexington Avenue
New York, N.Y. 10017
212-682-9494 1-800-223-7942

SOFTWARE/COMPILER

C SOFTWARE DEVELOPMENT
- Full C Compiler per K&R
  Inline 8087 or Assembler Floating Point
  Full 1 Mb Addressing for Code or Data
- MS DOS 1.1/2.0 Library Support
  Program Chaining Using Exec
  Environment Available to Main
- c-window™ C Source Code Debugger
- FAST 8088/8086 Assembler

COMBINED PACKAGE $199
c-systems
P.O. Box 3253
Fullerton, CA 92634
(714) 637-5362

C LIBRARIES + C WINDOWS
***** BEST YOU CAN GET *****
Over 325 richly commented tested functions
ALL SOURCE CODE! .... NO ROYALTIES!
**** 4 DISKS: 400 PAGE MANUAL ****
51 screen handling/graphics funct. $49.95
50 cursor/keyboard/data input $39.95
85 superior string functions $59.95
28 system status & control funct's $29.95
70 utilities/DOS/BIOS/time/date $59.95
42 printer control functions $39.95
No matter what ELSE you have, get these!!
Any 3 libraries ... $69.95 All 6 ... $99.95

C POWER WINDOWS C POWER WINDOWS
PROFESSIONAL WINDOW MANAGEMENT
overlays, borders, pop-up menus/help windows
status-line, color, highlighting, much more!
C WINDOWS: C complete source code........ $99.95

C WINDOWS plus 6 LIBRARIES all for $159.95

Entelekon
SOFTWARE SYSTEMS

Entelekon
12118 Kimberley
Houston, TX 77024
(713) 468-4412

DeSmet-C - $109
Full K&R C compiler, linker, librarian, fullscreen editor, profiler, overlays, example software, 140 page manual. BOTH 8087 and floating point libraries. OUTSTANDING PRICE/PERFORMANCE. Consistently rated 1st or 2nd in published benchmarks. PC/MSDOS. No royalties on generated code. Newsletter. Updates $20 ea. $109 shipping included. Symbolic (C source level). Debugger Option $50 extra. C Compiler for the Macintosh $150.
C WARE CORPORATION
Dept. PCTJ
P.O. Box C
Sunnyvale, CA 94087
(408) 720-9696

SOFTWARE DATA ENTRY

Data Entry System
Heads down data entry for the PC will allow fast and accurate preparation of data files for your mainframe. Features: Auto dup, verification, auto format change, table look-ups, range checks, batch totals, dup key, record insert, and much more. Ask us about our free 30 day trial period. Only $395.
COMPUTER KEYES
6519 193 SW
Lynnwood, WA 98036
(206) 776-6443

SOFTWARE DEVELOPMENT TOOLS

THE FORMS DESIGNER™
Attention IBM Pascal, FORTRAN users! Save time in designing formatted screen I/O. Interactive Forms Editor allows you to draw lines and boxes, define fields, and edit text. Access forms or read keyboard entry by writing only one line of code. Provides sequential data retrieval and storage. Requires 128K RAM. Only $275 complete. Demo and manual $35. Call or write:
BIT SOFTWARE
PO Box 619
Milpitas, CA 95035
(408) 262-1054

TURBO-PLUS™
Turbo-Plus™ is a set of external procedures crafted in assembler language and designed specifically for Turbo Pascal's interactive environment. Turbo-Plus™ includes: File Handle Disk Access Method, Instant Memory Mapped Text Display, Expanded Text Display. $34.95 plus $5 s&h.
NOSTRADAMUS
5320 South 900 East
Salt Lake City, UT 84121
(801) 261-0769

TEST FOR HARDWARE BUG
If your IBM PC is less than two years old, chances are it has a SUPERBUG in its hardware that will let you do something that should be impossible —to BEAT THE COMPUTER AT AMERICAN ROULETTE. To test your PC for this superbug, order a copy of the ROULETTE TEST PROGRAM by sending a money-order for $5.00 (Canadian) to:
Hardware Debugging Services (1985),
135 Cedargrove Court S.W.,
Calgary, Alberta, Canada. T2W 4T6.

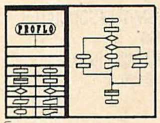
ASSEMBLE 3-6 TIMES FASTER
"FAST ASSEM 86" is much faster and easier to use than the IBM Assembler. Assembles 8K files 6 times faster. Editor, assembler and source always in RAM so you can go instantly from editing to assembling. No need to LINK since produces .COM (as well as .OBJ) files directly. Source compatible with IBM's important features. Supports 186, 286, 8087, 287 directly. X-ref. utility built in. Only $49, with .OBJ $99.
SPEEDWARE
118 Buck Circle
Folsom, CA 95630
(916) 988-7426

COLOR, CURSOR CONTROL
DATE/TIME, KEYBOARD, PEEK, POKE, COMMAND LINE ACCESS, and more available for IBM/MS FORTRAN, IBM/MS PASCAL, and GENERIC (MS) BASIC on PC/DOS and many MS/DOS machines. No redistribution fees. $47 U.S. MN residents add 6% sales tax. BARTSUBS by BARTSOFT
3210 Lake Shore Boulevard
Wayzata, MN 55391
(612) 473-4032

TURBO TUNE-up-19.95
Programs & Utilities adapted from TUNING YOUR TURBO PASCAL by Steve Wood, published by Osborne McGraw-Hill, copyright 1985-Osborne McGraw-Hill. Used with the permission of Osborne McGraw-Hill. The adaptions are solely the work of precision logic systems, and are not a publication of Osborne McGraw-Hill. Disk includes the complete Loan Amortization program, plus dozens of ready to use sub-programs. Bonus IBM-PC 'tools' not in the book. Add $3 S&H.
PRECISION LOGIC SYSTEMS
2727 E. 21st Suite 613
Tulsa, Oklahoma 74104
(918) 742-0979

TURBO EXECUTIVE™
Application Programmers: reduce development time! Combine the power of Turbo Pascal with a Multi Tasking, interrupt driven, Real-Time executive. Written in Turbo Pascal for control applications requiring efficient sharing of CPU by various tasks. Source code $59.95 + $5 s&h. Consulting and Custom versions available.
CLEAR CREEK COMPUTING
2107 Audubon
League City, TX 77573
(713) 332-9940

PROFLO BY STAR*SYSTEMS
THE flowchart drawing program for the software professional. Fast, responsive, easy to use. Menu-driven symbol selection. Full page flowchart display. Versatile text editor. Requires IBM PC or compatible with DOS, 128K, graphics display and Epson graphics printer. $49.95.



STAR*SYSTEMS
28 Topstone Drive
Bethel, CT 06801
(203) 744-0546

TECH BOOK

SERIOUS SOFTWARE DEVELOPERS
The SOFTWARE REVISION MANAGEMENT SYSTEM™ stores all versions of source/documentation in a single ASCII file. Allows retrieval of any version of source and application of changes, while recording when, why and where changes were made with no duplication of common code. MS/PC-DOS 2.0 $69.95 +. $5 s/h. (MN + 6%.) MC/Visa.
QUILT™ COMPUTING
7048 Stratford Rd.
Woodbury, Minnesota 55125
(612) 739-4650

DOS DEBUGGER WITH SOURCE
PC-TEST, a symbolic debugger so friendly you hardly need the manual. Multi-window memory display, text file browsing, port I/O, saved screens, 16 break points, loop breaker. 100% assembler, 64K, really flies. With optional commented source novices can learn/developers can customize. $39 or $129 (w/src) + $4 s&h.
AJM ASSOCIATES
10917 Northridge Square
Cupertino, CA 95014
(408) 996-7249

BASIC DEVELOPMENT TOOLS
The Programmer's Tool Kit is a collection of eight preprocessors and utility programs that speed development, documentation, and debugging. Code using labels instead of line numbers, remove line numbers, merge, cross-reference, compare, list, and compress. Write or call for full details.
DATASMITH, INC.
Box 8036
Shawnee Mission, KS 66208
Phone: (913) 381-9118

BASIC TOOLS
Screen Development Aid $29
- Create screens quickly (40/80)
- TEST screens prior to coding
- Automate screen functions from BASIC SORTSUB—Sorts any array $19
- Sort a 1000 element array in 3 secs.

Interp. & Compiled BASIC, DOS 1.1-3.0 IBM PC, PCjr, XT, AT and compatibles
DEPENDABLE SYSTEMS
P.O. Box 191090
San Diego, CA 92119
(619) 287-3391

C PROGRAMMERS
Professional Software Development Tools
— Cross Module Cross Reference
— Single Module Cross Reference
— FindText—all occurrances of a given string in specified files (like GREP)

/*ALL SOURCE CODE INCLUDED (Pure K&R C)*/
MSDOS, CCPM, UNIX, MacIntosh, others $165.00
SMITH & SMITH ASSOCIATES
P.O. Box 160
Hunt Valley, MD 21030
(301) 666-8129

** PASCAL'S FRIEND **
PASCAL'S FRIEND v2 contains source code for use with IBM PC Turbo Pascal: 1-2-3 style menu routines, keyboard handling, save & restore screens, read disk directory or any track & sector, system clock and calendar routines, write strings in any attribute, DOS function calls, MORE! $49.95 check, MO, Visa, MC
J.S. COMPUTING
815 N. 12th St. Suite 5
Allentown, PA 18102
(215) 821-9020

BTREES IN C WITH SOURCE
The Softfocus B-Tree Library is a complete set of data management functions written in K&R C. We support random and sequential access, multiple indices per data file and up to 16.7 million keys per index. Also, our software is customizable because we supply full source code. Manual & examples included.

B-Trees
$77.00

SOFTFOCUS
1277 Pallatine Drive
Oakville, ONT. CANADA L6H 1Z1
(416) 844-2610

SOURCE DEBUGGER FOR LATTICE C
The MSD DeBugger™ is the last, and perhaps final, word in programming assistance for Lattice C users. C Debugger produces a high-level view of C programs via function names line numbers, variable names and C data types, plus a low-level view of machine addresses and instructions for testing assembler language functions. More features include:
- All documentation is prepared for programmers.
- Online help screen throughout the process.
- Capability to single step through your program.
- Set break points, examine registers and variables.

$165.00 + $3.50 shipping VISA/MC

Cdebugger
a high level language debugger

MSD
214 1/2 West Main St.
St. Charles, IL 60174
(312) 377-5151

SoftRoms
High Performance Graphics Tools. FAST; 200K pixels/sec avg. TIGHT; 14K. COMPLETE; 2/3D-Poly Lines, Bit Blocks, Windows, Fill, Arc, Icon, Character, String, Macros over 100 functions. Plus; assembler hooks, multiple language & hi-res boards, icon editor, memory draw & print, mouse & keyboard touch, type faces. Basic kit $70.00.
AFFINITY, INC.
126 Normal Ave.
Buffalo, New York 14213
(716) 882-5077

CROSSREF/BASIC & PASCAL
Debug programs, customise listings, control evasive variables! Our utilities will list your source, cross-reference variables in sorted order, and list procedure/function names. Available for Microsoft® Basic and Turbo Pascal® at $29.95 each or receive both for $49.95. All orders add $4.00 S&H. Check or Money Orders (Inquiries invited) to:
EXECUTIL
PO Box ### (12516)
Albany, N.Y. 12212

C TOOLS—MAKE
MAKE automatically keeps track of the dependencies between modules and include files. When you edit a file, you need only invoke MAKE to make sure that all the necessary modules are correctly compiled and linked and that object libraries are updated. Includes other useful tools. $40.00 check or COD.
INFOSOFT, Inc.
9020 Perryvale Rd.
Balto., MD 21236
(301) 256-2631

SUPERTOOLS™ FOR TURBO
With Supertools™ Turbo Pascal™ you can write programs that (pop up) from with Lotus 1-2-3,™ dBase,™ etc. In addition we've included Morse Code MACS™ a resident keyboard macro-processor.
Routines # included let you...
- Sidekick-like™ resident program.
- Redefine your keyboard
- Save and restore screen windows
- Write strings directly to video
- Read strings from the screen

MACS™ keyboard enhancement features:
- Menu driven
- 200 simultaneous key defined
- 1000 characters maximum per macro
- Resident macro editor
- Load and save from within program
- Redefine special keys (on the fly)
- Predefined macros for Turbo and DOS

Additional routines include: time & date access, program execution return, DOS memory allocation. Supertools. Send $54.99 to:
SUNNYHILL SOFTWARE
13732 Midzale North Suite 206
Seattle, WA 98133
(206) 367-0650

THE HAMMER: TOOLS INC.
More than just BIOS/DOS access, THE HAMMER also provides screen attribute control, 123-like menus, easy data entry & verification of dates, numbers, & strings, AND MORE. Super data entry routines work together in single or multi-field (FULL-SCREEN EDITING) mode. This is NOT just another general purpose library. Supports: CI-C86, Lattice, Desmer, and new microsoft V3 C compilers. $195 wtih source & manual. Call or Write:
OES Systems
1906 Brushcliff Rd.
Pittsburgh, PA 15221
412-243-7365

SOFTWARE/ ENGINEERING

smARTWORK®
Printed-circuit artwork editor for double-sided boards up to 10" by 16", runs on an IBM-PC. Color display allows complete interactive control over the placement and routing process. 2X artwork can be made on a dot-matrix printer or pen-and-ink plotter, $895. Write or call for a brochure.

WINTEK

WINTEK Corporation
1801 South Street
Lafayette, IN 47904-2993
(317) 742-8428

METAL FABRICATORS
PC/Cutlist takes input from your bill of material—Detail drawing and calculates the best cutting combination for any length stock and prints a shop ready cutting list and scrap report. Also an optimization feature finds best mult length for mill orders. Price $300. Demo Disk $25.00
THE JOSEPH ALBERT CO.
P.O. Box 611
Blue Island, Illinois 60406
(312) 349-9032

KALMAN FILTER PRIMER
Learn Kalman Filter with this software, document, and support package. Easy-to-use programs provide practical hands on experience. Unique step-by-step approach designed for self-study. Many numerical examples with straightforward explanations. For engineers, scientists, programmers, and managers.
KALMAN FILTER PRIMER
22338 Lull Street
Canoga Park, CA 91304
(818) 716-0816

MEGA MATH
A library of over 40 assembly language subroutines for fast numeric calculation on IBM PC/XT/AT and compatibles includes matrix and vector routines. Fast Fourier Transform, convolution solution of linear equations. Use with IBM and MS Fortran, C, PASCAL, BASIC and assembler. One complete library $99, each additional, $49. VISA/MC/AMEX.
MICRAY ELECTRONICS LTD.
Bay 1, 4001A-19th St., NE
Calgary, Alberta, Canada T2E-6XB
(403) 250-1437

STEAM
STEAM computes the thermodynamic properties of water over a wide range of values commonly used in applied thermodynamics. The FORTRAN package includes an interactive "stand-alone" program and code in object form, callable as a subroutine. Specify your FORTRAN compiler. $295. (Calif. + Tax).
SOFTset, INC.
10889 Wilshire Boulevard, Suite 810
Los Angeles, CA 90024
(213) 824-1904

TECH BOOK

FFT FOR PC 8087
The speedy VECTOR87 library is coded in MASM and includes forward/inverse FFT, convolution, and array processing routines. 1024 point FFT takes 1.2 seconds real, 2.2 seconds complex. Use with PC, PC/AT, Fortran (MS 3.2/IBM 2.0), C (Lattice 2.1) and Pascal (IBM 2.0).
$49 US postpaid by check, money order, VISA.
VERTORPLEX DATA SYSTEMS LTD
P.O. BOX 138 Station M
Calgary, Alberta, Canada T2P 2H6
(403) 248-1250 after 7pm MST

SOFTWARE/ EXPERT SYSTEMS

BUILD EXPERT SYSTEMS FAST
Use EXPERT-EASE™ to build expert systems in any field w/o prior experience. PC Mag: "One of 1984's best products". This AI breakthrough is now the top selling expert system shell (generator). For IBM PC/XT, clones, DEC Rainbow, 128K + RAM. Full function demo version w/manual-$60. Full credit applied to complete software system-$495. More info: contact,
EXPERT SYSTEMS, INC.
868 West End Ave., Suite 3A
New York, NY 10025
(212) 662-7206

SOFTWARE GENERAL

RENT PUBLIC DOMAIN SOFT/W
Rent our PC Disk Libraries for 7 days and copy them yourself. Hundreds of useful business, games and utilities FREE FOR THE TAKING!

IBM-PC SIG-220 +0220 Disks	$230.00
IBM-PC "Blue" 80 +080 Disks	$85.00

Rental is for 7 days after receipt, 3 days grace to return. Most credit cards accepted.
NATIONAL PUBLIC DOMAIN
SOFTWARE CENTER
1533 Avohill Drive
Vista, CA 92083
(619) 941-0925. Info and orders. (619) 727-1015. 24 hr. orders.

C Users' Group

PUBLIC DOMAIN SOFTWARE IN C
Over 45 volumes of public domain software in CP/M & MS-DOS formats.
- editors & compilers
- text formatters
- communications packages
- many UNIX-like tools

Write or call for more details.
THE C USER'S GROUP
415 E. Euclid Box 97
McPherson, KS 67460
(316) 241-1065

SOFTWARE GRAPHICS

TEKTRONIX 4010 EMULATION
High resolution screen/printer graphics for the IBM PC. Full interactive capabilities with file transfer and cross-hair control. Utilizes PLOT 10, protocols. OFF-line review of graphics output. Supports IBM color/graphic or Hercules cards. Easy to use 4010 emulation at an affordable price. $80.00. Demo disk $5.
Technological Systems Group
5044 Haley Court
Lilburn, GA 30247
(404) 923-4980

GRAPHICS LIB FOR TECMAR
TEK-MAR lets you do high-res graphics on your TECMAR Graphics Master. TEK-MAR is a library for use with MS Fortran. Features windowing, viewporting, clipping, axis rotation. Similar to Tektronix graphics. Includes screen dump/restore, Epson screen print, support for Hewlett Packard, Western Graphtec plotter. Requires MS-FORT 3.20, 320K, GMDEV.SYS (5.2). Price: $195.
ADVANCED SYSTEMS CONSULTANTS
18653 Ventura Boulevard, Suite 351
Tarzana, California 91356
(818) 990-4942

MetaWINDOW™
Advanced graphics toolkit provides Xerox Star/Apple Macintosh style graphics on your IBM PC. Supports most popular graphics cards. Allows you to create pop-up menus, windows & icons; use proportionally spaced fonts; rubberband & drag lines, text or bitmap images; supports mouse/cursor tracking. Tightly optimized for use with Turbo Pascal, IBM Pascal, C, Fortran & Compiled Basic.
METAGRAPHICS SOFTWARE CORP.
444 Castro Street, Suite 400
Mountain View, CA 94041
(408) 438-5352—Telex: 4998586 METAGRAF

SOFTWARE/ INTERPRETERS

C-Terp™-C Interpreter
A COMPLETE DEVELOPMENT ENVIRONMENT. Full K&R language and libraries; Complete built in screen editor;
compiler compatible, supports object library compiled with C86 or Lattice C or assembler;
Multi-module support, includes global search; High-level symbolic debugging; Batchmode; Incredibly fast linking and semi-compilation; Runs on IBM-PC Dos 2.X, 3.X, 192K and up. Price: $300 (Demo plus manual $45) MC, VISA. See our display ad for more info.
Gimpel Software
3207 Hogarth Lane
Collegeville, Pa 19426
(215) 584-4261

SOFTWARE/LANGUAGES

IBM PC PASCAL/DOS 3.0/1
For the serious PASCAL and DOS user. EXAMPLES PLUS provide a rich set of examples and directly usable PASCAL function/proc code for PASCAL with DOS 2.0/1. EXAMPLES PLUS 2 provides additional info for PASCAL 2 with DOS 3.0. EXAMPLES PLUS 3 shows the differences and glitches in DOS 3.0/1, plus info on TopView, networking, extended graphic board and the AT hardware. Priced at $30 each, 2 for $48, 3 for $60. (CA add 6%).
JATO INC.
24861 Alicia Parkway, Suite 272-B
Laguna Hills, CA 92653

PASCAL-TO-C-TRANSLATOR
Convert UCSD Pascal, MT+Pascal, and others to K&R C. Handles nested procedures, intrinsic functions, separately compiled units and modules, all data types including long integers. Requires 256K IBM PC/XT/AT. Send for free samples or send us up to 500 lines of Pascal and we will convert it for you FREE. Licensed source and executable code $5,000, conversions 25 cents/line.

TGL INC

TGL INC.
4400 Sulphur Springs Road
Corvallis, OR 97330
(503) 745-7476

SOFTWARE/NETWORK

IBM PC NETWORK
Products: electronic 'chat' system, network driver/ioctl functions, comm port sharing for modem pooling; End User services: custom software development, consultation for larger user communities (including capacity planning), remote support, joining of PC networks together forming a single wide area network

PC Network Specialists, Inc.
13109 Chandler Drive
Dallas, TX 75243
(214) 783-6313

SOFTWARE/OPERATING SYSTEMS

pcSHARE MULTI-USER O/S
pcSHARE allows your IBM-XT or compatible to support up to 4 users running 123, dBASE, WordStar, etc on inexpensive serial CRTs. For the software developer, pcSHARE efficiently runs compiled Basic, Pascal & C programs with full DOS 3.0 record locking. No risk 30 day money back guarantee.
DIGITROL COMPUTERS, INC.
440 Phillip Street
Waterloo, Ontario, Canada N2L 5R9
(519) 884-4541

CX/PC C EXECUTIVE
Real-time, priority task scheduler. Functions to queue inter-task messages and pointers. Example tasks. Interrupt-driven timer, serial and parallel port interfaces. Runable demonstration programs. DeSmet C source included; no license for generated programs. Described in June '84 IEEE Micro article. $75. Brochure.
INTR-SOFT CO.
P.O. Box 351
Bedford, MA 01730
(617) 369-6242

SOFTWARE/SCIENTIFIC

HP-PC HYPER-CALCULATOR
Activate emulation of an HP-11C programmable scientific calculator from any program, and return with the results. Features program and data storage on disk, indirect addressing, alpha-numeric messages, printing, on-line HELP, 100 registers, and 1,000-step programs for solving roots, numerical integration, complex numbers, matrices, base conversion, time/money problems. $39.95.
SUNDERLAND SOFTWARE ASSOCIATES
P.O. Box 7000-64
Redondo Beach, CA 90277
(213) 540-2567

SOFTWARE/SECURITY

SECURE AT/XT/PC
Control system access, data access! FiXT/S. Control system boot for most popular XT/PC hard disk controllers. Vfeature for AT-and-XT-compatible HD controllers segments hard disk by volumes, controls access with passwords, supports hard disk expansion. $70-$130 + $3 shpg. plus CA tax.
Golden Bow Systems
P.O. Box 3039
San Diego, CA 92103
(619) 298-9349

SOFTWARE/SERVICES

TAPE TO DISK CONVERSIONS
Convert any 9 track magnetic tape to or from over 200 formats including 3½", 5¼", 8" disk formats and word processors. Formats available include IBM-PC, Apple, Altos, TRS 80 8" CP/M, Displayer., IBM Sys/??. Macintosh, Wang, Lanier, OS/6 and 300 more. Disk to Disk conversions also avail. Call for more info.
PIVAR COMPUTING SERVICES, INC.
47 W. Dundee Rd.
Wheeling, IL 60090
(312) 459-6010

TECH BOOK

Software/Spreadsheet

FORMAT—1-2-3
- **Loads ANY ASCII data into Lotus™!**
- 1-2-3 and Symphony Utility!
- Send spreadsheets using E-Mail!
- On-Line Help System!
- Runs on PC, XT, AT, and lookalikes!
- Also TI Professional, AT&T
- Extended Warranty Available!
- DOS 2.0 + with 64K, 1 drive
- $150, MC, VISA, CK, COD

RELIABLE SOFTWARE COMPANY
P.O. Box D
Titusville, FL 32781
(305) 267-2043
(800) 235-6646 Ext 581 (nat)

Software/Stock Analysis

STOCK OPTION TOOL KIT
Use Monte Carlo analysis and random walk theory on stock option strategies. The kit has 50 pages of documentation, bibliography and 11 programs to evaluate covered calls, naked puts and calls, and various spreads. Price $199. Demo disk with one program $25. IBM PC, PC XT, 128K, DOS 2.0+ check or MO.
EXATECH SOFTWEAR
939 Pulpit Rock Circle North
Colorado Springs, CO 80907
(303) 599-0280

PC-STOCK GRAPHIC ANALYSIS
Graphic trend analysis aiding in technical stock, bond, and commodity analysis. Includes "ON BALANCE VOLUME" and moving average. Ideal for the small investor. PC-STOCK comes complete with source code and examples. Requires IBM-PC, 128k, DOS 1.1-2.1 and hi res graphics. $30 + $3 shipping.
SUNRISE ENTERPRISES
P.O. Box 2610
Goleta, CA 93118
(805) 968-4170

Software/Typesetting

HIGH-TECH TYPESETTING
Transmit your text via toll-free lines directly to our fully automated typesetting system. $2. per K characters with a $5. minimum. Same day service. 200 typefaces in sizes up to 72 point. Send $15. + $3. shipping for our 220 page guidebook, or call toll free and use your MC, VISA or AMEX.
INTERGRAPHICS INC.
106A South Columbus Street
Alexandria, VA 22314
(800) 368-3342 or (703) 683-9414 in DC area.

Software/Terminal Emulation

VT52/102/TEKTRONIX 4010 PLUS FULL FILE TRANSFER
ZAP does amazing emulation of the above terminals and includes: all VT102 attributes to 132 column, nested command files, keyboard reconfiguration, XMODEM, kermit, and text file transfer, online help, text and graphics, translate bytes received, DOS commands, AND a sophisticated, general purpose package to interact with any computer, from PCs to super-mainframes. Requires IBM PC, AT, or compatible, and 128K.

Only $65 plus $3 shipping. Full refund if not satisfied in first 30 days.

Solution Systems™

SOLUTION SYSTEMS
335-TN Washington Street
Norwell, MA 02061
800-821-2492, 617-659-1571

Software/Utilities

DISK MECHANIC
THE ULTIMATE Floppy Disk Backup & Repair Utility. Can back up ALL software protected disks written on the IBM PC. Works manually or automatically. Files or sectors can be restored, searched, examined & changed. Checks disk drive speed. req. IBM PC, DOS 1.1 (128K)/2 (192K) + 64K if only 1 floppy drive. $73 ppd. USA
MLI MICROSYSTEMS
PO BOX 825, Dept. TB
Framingham, MA 01701 USA
(617) 926-2055 for info MC/VISA

DIGICON PRINT PACKAGE
Want SIDEWAYS printing for SPREADSHEETS? How about BANNNER, PAMPHLET, LETTER, and BOOKLET printing too? Variety of fonts. Graphics characters. Character editor to MAKE YOUR OWN FONTS or SYMBOLS! Quick printer setup program too. Easy menus, online help. IBM/Epson printers, DOS, 128K. $49.95 (PA+6%).
DIGITAL CONCEPTS, INC
POB 8345 Dept. A
Pittsburgh, PA 15218
(412) 823-8314

NOBLINK™
ErGoNoMiCs! Eye strain? With just a keystroke you can change your blinking cursor into a friendly solid block reverse video cursor. If you wish you can change it back and forth (Blink/No blink) right in the middle of a program, such as your word processor. Give your eyes a rest. $29.95 plus $5 s&h.
NOSTRADAMUS
5320 South 900 East
Salt Lake City, UT 84121
(801) 261-0769

IBM USES CPM DISKS!!
Uniform lets you use CPM disks in your IBM B: drive while running any PCDOS/MSDOS program. Read and write to the CPM disk directly or transfer files to IBM format. Over 70 formats also 8" and 96 TPI. IBM PC, XT, AT. Call for details. Dealers invited. (Not for Apples). $59.95 + $5 S&H. Uniform by Micro Solutions.
BLUE HERON
RR 1, Box 128 E
Maple Park, IL 60151
(815) 895-5602

Hard Disk DIRECT ACCESS™
The ULTIMATE utility for any hard disk user. DIRECT ACCESS is a powerful hard disk management tool. Organizes your software programs into a "user defined" menu system. Features attractive screen display, rapid access into an application using a single keystroke & easy to learn. Also handles batch files & complicated DOS commands. Order today. Only $59.95 + $5.00 shipping & handling. MC/VISA.

DELTA TECHNOLOGY

DELTA TECHNOLOGY
P.O. Box 1104
Eau Claire, Wisconsin 54702
(715) 832-0958

AVOID FRUSTRATION!
PC-CATLG—easy to use, reliable, Diskette Management Utility. Directory info for EACH file is read/stored in a master catalog with a 40 char user comment. Functions: ADD/CHG/DEL/SRCH/UPD/DISPLAY/PRT/LABELS. Chg VOL labels. Req. DOS 1.1/2.X, 256K, 80 mono/color, 2 DSDD drvs (SSDD add $3), parallel printer opt. $25.
Oregon-microTech
1717 Magnolia Ave.
Medford, OR 97501
VISA/MC (503) 773-8327

DBASE II dBDecode
dBDecode is a utility program that converts programs that have been compiled with the D-BASE II DBCODE compiler, back into the original source code format. This powerful utility among others, accounts for identification of IF-THEN-ELSE-ENDIF and DO-WHILE-ENDDO constructs during source code generation. A must for DBCODE users. $69.95.
DATAGLOBE SYSTEMS
30 Burnhill Rd., Suite 803
Scarborough, Ontario, Canada

THE NIBBLER™ 2.0
Powerful disk, file & memory mgmt. utility for programmers & DP professionals! view, encrypt, edit, recover, copy, delete, hide & rename files. View & Edit Memory. Hex/ASCII search in files & memory. File maps; Label/ Directory editing; windows; context-sensitive HELP; Transfer CPM & p-system data to DOS; Ram & Fixed Disks! written in assembly for super speed! $69.95 + $2.50 S&H. (CA add 6%) check/Visa/MC.
TACHYON SYSTEMS INC.
2725 Congress St. Suite 2H
San Diego, CA. 92110
(619) 574-1666

FKEY NOW AVAILABLE
The famous FKEY key processor is now running on PC DOS/MS DOS. FKEY allows the redefinition of each key and key macros. No royalties on the run time package for product developers. Attractive dealer discounts and dealers are invited. Call or write for more information about FKEY. $39.95.
DITj
P.O. Box 60727
Sunnyvale, CA 94088
(408) 737-9000

SPEED UP YOUR DISPLAY!
FANSI-CONSOLE™: faster screen writing thru PC-DOS or BIOS (1.3-3.0+ times), larger X3.64 subset than ANSI.SYS, VT100/52 keys, more type-ahead, auto inactivity screen disable, eliminate scroll blink for some adaptors, window support, keyboard breakpts. Software & printed manual $50, or shareware disk $25.
HERSEY MICRO CONSULTING, INC.
P.O. Box 8276
Ann Arbor, Michigan 48107
(313) 994-3259

SMART FILE COMPARISON
CMP finds changes, including **blocks** of lines that have **moved** between two ASCII files. It finds lines that have changed **within** a block that has moved. 20,000 line maximum, skip blanks, output to file. PC-DOS or CP/M-80, Ver 2.13 $89 (+ sales tax in CA) Visa/MC.

WHITE HERON CORPORATION

White Heron Corporation
Mail Order Dept.
26739 Basswood Ave.
Rancho Palos Verdes, CA. 90274
(213) 378-4429

ALL DISK COMMAND SYSTEM
Impressive FILE HONEY™ system gives total access to subdir with 4 arrow keys. Sharpen PCs/XTs. Pampers today's users. Sorts, organizes, ID's, locates, scans-FAST-without memorizing subdir names. New tree access! Sophisticated wildcards & dates. Extra print features you like! Handsome folio $50—Demo $15. 128K/DOS 2.X.
THE JOE PHILLIPS COMPANY
9071 Metcalf, Suite 145-C
Shawnee Mission, KS 66212
(913) 341-3645

ASSEMBLY LANGUAGE PROS!
Sick and tired of those sloppy, misaligned listings of your masterpieces? ASMPRT will let *you* control where the tab stops are set for a slick, professional-looking program listing. Also allows you to compress the listing for those 80-column printers. Prints all other ASCII text files, too! Only $29.95.

Ci
Childs Systems Inc. USA

CHILDS SYSTEM, INC.
P.O. Box 45212 Dept. 485
Baton Rouge, LA 70895
(504) 291-4167

TECH BOOK

WHAT IBM LEFT OUT . . .

The enhancer offers: fast flicker-free scroll • display review • pause at screen full • choose color for input • solid block cursor • oversized editable writable keyboard buffer displayed on screen • alarm clock with time display and chimes • menu entry of special characters. IBM PC & XT $50 check/MO Colo. res. add 3.6%.
GENERAL SOFTWARE
P.O. Box 3272 Dept 12
Boulder, CO 80307-3272

COMPUTER SYSTEM LOG

Monitor activities on your computer with SYSLOG. The System Log Utility is a command driven nonresident DOS extension that allows automatic task recording. With SYSLOG, all of your computer time is noted, saved, and categorized. Reports printed for tax purposes or for monitoring PC usage. Only $34.95 + Sh. VISA/MC accepted.

SYSLOG
SYSTEM LOG UTILITY

Apex Resource, Inc.
23 Christine Court
Stormville, NY 12582-9612
(914) 221-2611

UNIVERSAL CROSS REFERENCER

One Cross Referencer does it all for BASIC, C, Pascal, FORTRAN, Modula2, COBOL, DBASE-II, Macro Assembler, etc. User-defined page titles, page numbers, file date/time. Unlimited symbol length, adjustable output margins, compressed print. Multiple input files, Full directory and path support. DOS 2.x, IBM PC/PC-XT, 128k, $39.95.
DALSOFT Engineering
3565 High Vista
Dallas, Texas 75234

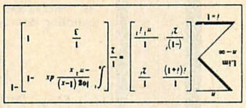
DOS PATH COMMAND FOR DATA

Now Lotus, c-Base, WordStar etc. can access data files no matter where they're located on your hard disk, floppies or RAM disk. D-PATH lets you organize your directories conveniently for you, not just your programs. Requires only 4K. Works on MS-DOS 2.0, 3.0 $25 + $25/h Visa, MC, company P.O., check.
PERSONAL BUSINESS SOLUTIONS INC.
P.O. BOX 757 DEPT. T
Frederick, MD 21701
301-865-3376

SOFTWARE/WORD PROCESSING

FORMATH™ TEXT-FORMATTER

For technical texts. Sizes & places equations, matrices, ratios, integrals, big symbols, footnotes. Macros, multiple fonts, 158 Greek/math symbols. Automatic hyphenation, section/equation/reference numbering, indexes, table of contents. Dot-matrix, daisywheel, laser printers, plain/graphics monitors. $400. $50 for demo.



SHANTHA SOFTWARE, INC.
50 West 97th St. Room 11N
New York City 10025
212-222-7647
Touch-tone Toll-free: 950-1088-wait-FORMATH

VOLKSWRITER DELUXE

Features TextMerge™ for customized mailing lists; multi-lingual, engineering & scientific symbols; horizontal scrolling to 250 characters; document size 1 megabyte; on-screen page endings, underline, boldface, double-strike, strikethrough, multiple fonts & colors—fewest keystrokes per function. Only $295.
LIFETREE SOFTWARE INC.
411 Pacific Street
Monterey, CA 93940-2790

PC BUYERS GUIDE

THE INDEPENDENT GUIDE TO PRODUCTS AND SERVICES FOR IBM Personal Computers. This master directory lists, describes, rates for ease of use, and tells you about the latest in software, hardware, accessories, supplies, books, services, et al. Everything for you and your IBM PC. It even includes an Index of Computer Camps for both adults and children. From the publishers of PC. An ideal gift at only $7.95 (+ $1.55 shipping/handling) to U.S. destinations; and $11.00 elsewhere. Residents of CA, CO, CT, DC, FL, IL, MA, MD, NJ, and NY add applicable sales tax. Checks or MasterCard or VISA acceptable (with number, expiration date and signature). U.S. funds only. Mail to:
PC BUYERS GUIDE
CN 1914
Morristown, NJ 07960

TECH MART

FORTRAN or PASCAL PROGRAMMER?

READ THIS AD

GRAPHICS
- Text/graphics generics
- 2D interactive
- 2D plots (full support) including auto-scaling, auto-axis generation, auto-labeling, tabular/log/parametric curves, splines, bars, pies, you name it, we have it.
- 3D Plots incl. 2 hidden line removal options—not just old-fashioned wire frame.

PEN PLOTS
- Standard plotter primitives plus
- FULL 2D support plus
- Interface to screen graphics,
- Limited 3D.

Clear and complete documentation.

GRAFMATIC	$135.
PLOTMATIC	135.

MICROCOMPATIBLES
301 Prelude Drive
Silver Spring, MD 20901
(301) 593-0683

CIRCLE 400 ON READER SERVICE CARD

Now With Windowing!
$49.95 Basic Compiler

MTBASIC

Features:

Multitasking	Windowing
Handles interrupts	Interactive
Fast native code	Compiles quickly
Floating point	No runtime fee

MTBASIC is a true native code compiler. It runs *Byte's* Sept. '81 sieve in 26 seconds; interpreters take over 1400 seconds! Because MTBASIC is multitasking, it can run up to 10 Basic routines at the same time, while displaying ten separate windows. Pop-up/down menus are a snap to implement.

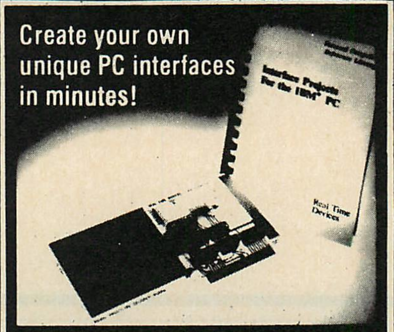
The MTBASIC package includes all the necessary software to run in interpreter or compiler mode, an installation program (so *any* system can use windowing), three demonstration programs and a comprehensive manual.

AVAILABLE for CP/M (Z-80), MS-DOS, and PC-DOS systems.

ORDERING: Specify format when ordering. We accept Visa, MC, checks and COD. Send $49.95 plus $3.50 shipping and handling ($10 overseas) to:

SOFTAID, Inc.
P.O. Box 2412 Columbia, MD 21045-1412
301/792-8096

CIRCLE 399 ON READER SERVICE CARD



The PD100 allows rapid development of specialized PC interfaces. It features a buffered data bus, switch-selectable address decoder, and a large prototype area (up to 40 IC sockets). The 116-page manual covers basic interfacing concepts and details implementing A/D, D/A converters, I/O ports, connection of transducers and dozens of useful circuits.

PD100 w/manual - $99 plus $3.50 P&H	REAL TIME DEVICES
Manual only - $20 Postpaid	1930 Park Forest Ave
PA residents add 6%	P.O. Box 906
Check, MasterCard, or VISA	State College, PA 16804
	(814) 234-8087

PERFECT COPIES GUARANTEED

Why risk duplicating your important programs on your computer, when *our* equipment is designed solely to duplicate disks and verify their perfection 100%?

Over 600 formats including Commodore 64. Plus serialization, copy protection, labeling, packaging, shrink-wrapping and *fast*, personalized service.

Western Transdata Inc.
DISK DUPLICATION

1701 E. Edinger Ave.
Building A7
Santa Ana, California 92705
(714) 547-3383 (Collect)

CIRCLE 396 ON READER SERVICE CARD

FAST SCREEN OUTPUT FOR TURBO PASCAL

FASTSCREEN ™
is a set of **inline** assembler and Pascal procedures for Turbo Pascal users.
- Display an entire screen or window almost instantly
- Process multi-field input screens that give your user full cursor control
- Sample program uses Conway's **LIFE** to illustrate use of procedures
- All source code included
- IBM PC/XT/AT and true compatibles

29[95] VA residents add 4% sales tax

TECHNISOFT
1710 Allied Street, Suite 37
Charlottesville, VA 22901
(804) 979-6464

Turbo Pascal is a registered trademark of Borland International

CIRCLE 397 ON READER SERVICE CARD

QNX

NEXT DAY FREE DELIVERY *

QNX 1.2 Development System	595
QNX 2.0 Networking Version	1190
QNX Networking Board	425
SOFPLAN Multi-User Spreadsheet	200
ZIM Multi-User Database	1095

8087

8087-3 (5MHZ) Co-Processor Kit	159
8087-2 (8MHZ) Co-Processor Kit	315
80287 (AT) Co-Processor Kit	280
FORLIB: Fortran Scientific Subroutines	99

* QNX ORDERS OVER $500

kanatek
MICRO
Moodie Drive High Tech Park
105-215 Stafford Rd. Ottawa, K2H 9C1
(613) 726-1907
Quantity discounts Dealers welcome

CIRCLE 398 ON READER SERVICE CARD

ICs PROMPT DELIVERY!!!
SAME DAY SHIPPING (USUALLY)

OUTSIDE OKLAHOMA: NO SALES TAX

ADD 640 Kbyte TO PC/XT 8-SLOT MOTHERBOARD CONSUME NO EXPANSION SLOTS: $78.50

QUANTITY ONE PRICES SHOWN

8087-3	MATH		$105.00
8087-2	COPROCESSORS		140.00
DYNAMIC RAM			
256K	256Kx1	120 ns	$ 4.49
256K	256Kx1	150 ns	3.25
64K	64Kx1	150 ns	.99
EPROM			
27C256	32Kx8	250 ns	$15.99
27256	32Kx8	250 ns	9.10
27128	16Kx8	250 ns	3.47
27C64	8Kx8	200 ns	7.85
2764	8Kx8	250 ns	2.50
2732A	4Kx8	250 ns	2.75
STATIC RAM			
6264LP-15	8Kx8	150 ns	$4.99
6116LP-3	2Kx8	150 ns	1.95

OPEN 6½ DAYS: WE CAN SHIP VIA FED-EX ON SAT.

NO EXTRA COST ON F-EX SAT DELIVERY

MasterCard/VISA or UPS CASH COD
Factory New, Prime Parts µP∞
MICROPROCESSORS UNLIMITED
24,000 S. Peoria Ave., BEGGS, OK. 74421 **(918) 267-4961**

Prices shown above are for July 2, 1985

Please call for current prices. Prices subject to change. Please expect higher or lower prices on some parts due to supply & demand and our changing costs. Shipping & insurance extra. Cash discount prices shown. Orders received by 6 PM CST can usually be delivered to you by the next morning, via Federal Express Standard Air @ $6.00, or Priority One @ $15.00!

PC THE BUYERS GUIDE

THE MASTER DIRECTORY OF PRODUCTS FOR THE IBM PC, PC XT, PCjr AND MOST COMPATIBLES!

AVAILABLE AT YOUR LOCAL NEWSSTAND AND COMPUTER STORE.

INDEX TO PRODUCTS

Does your ISAM run on IBM, APPLE, DEC and AT&T computers? c-tree does, and you only BUY IT ONCE!

The company that introduced micros to B+ Trees in 1979 and created ACCESS MANAGER™ for Digital Research, now redefines the market for high performance, B+ Tree based file handlers. With c-tree™ you get:

- complete C source code written to K&R standards of portability
- high level, multi-key ISAM routines and low level B+ Tree functions
- routines that work with single-user and network systems
- no royalties on application programs

$395 COMPLETE

Specify format:
5¼" PC-DOS 3½" Mac
8" CP/M® 8" RT-11

for VISA, MC or COD orders, call
1-314-445-6833

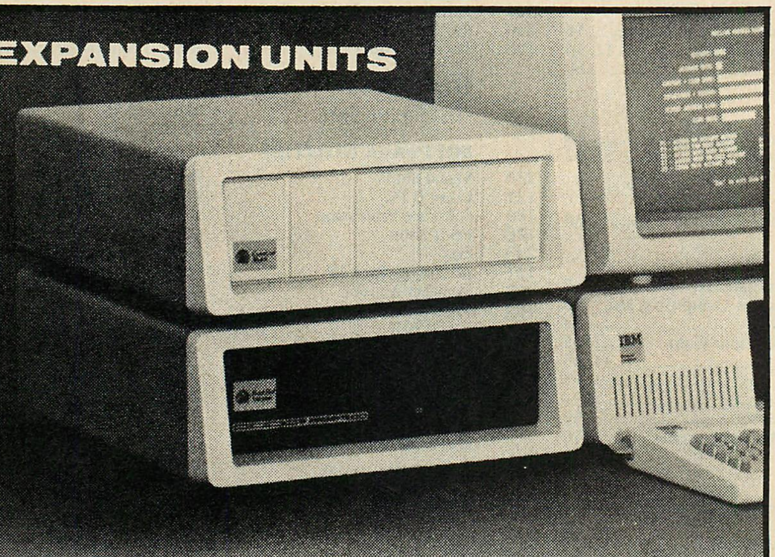
c-tree™
BY FAIRCOM

2606 Johnson Drive
Columbia MO 65203

Access Manager and CP/M are trademarks of Digital Research, Inc. Apple is a trademark of Apple Computer, Inc. c-tree and the circular disc logo are trademarks of FairCom IBM is a trademark of International Business Machines Corporation DEC is a trademark of Digital Equipment Corporation ©1984 FairCom

CIRCLE NO. 119 ON READER SERVICE CARD

PC EXPANSION UNITS



HARD DISK SUBSYSTEM

- 10 OR 20 MEGABYTE CAPACITY
- DOS 2.0 COMPATIBLE
- SYSTEM INCLUDES DRIVE INSTALLED IN CABINET WITH POWER SUPPLY, INTERFACE CARD FOR INSTALLATION IN PC AND CONNECTING CABLES
  10 MB - $1595 20 MB - $1895

BUS EXPANSION MODULE

- DOUBLES THE EXPANSION BOARD CAPACITY OF THE PC
- INTERFACES TO PC THRU SIGNAL CONDITIONED CABLE ASSEMBLY WHICH OCCUPIES A SINGLE SLOT IN PC
- SYSTEM INCLUDES CABINET WITH 6 SLOT BACK PLANE AND 4 OUTPUT POWER SUPPLY, INTERFACE CARD AND CONNECTING CABLE - $695

DESERT TECHNOLOGIES
P.O. BOX 41628
TUCSON, AZ 85717
(602) 882-1741

CIRCLE NO. 138 ON READER SERVICE CARD

MBASIC Programmers

Save **Time** & **Money** with

HELP MESSAGE SYSTEM™

Minimize documentation! Create and edit, context sensitive, interactive help and/or tutorial functions. Turns your text information into callable libraries. Overhead to your program is insignificant.

B-PLUS TREE SUB-SYSTEM™

Write your program without worrying about the handling of index files. Input/output is almost instantaneous, independent of file size. Direct and sequential accessing, variable length keys, multiple non unique keys and multiple keys per record.

SCREEN BUILDER SYSTEM™

Using either the built in editor or your word processor, create edit fields of any length anywhere on the screen with full screen editing within each field and full screen movement from field to field. The generated code is ready to insert in your program.

EZ SCREEN™

An Assembler language routine embedded in a basic subroutine to perform screen handling and windowing functions when called from a Basic program.

Priced As Low As **$59** Each

30 DAY $ BACK GUARANTEE
NOT COPY-PROTECTED
NO ROYALTY

Any 1 Program $79
Any 2 Programs $138 ($69 each)
Any 3 Programs $198 ($66 each)
All 4 Programs $236 ($59 each)

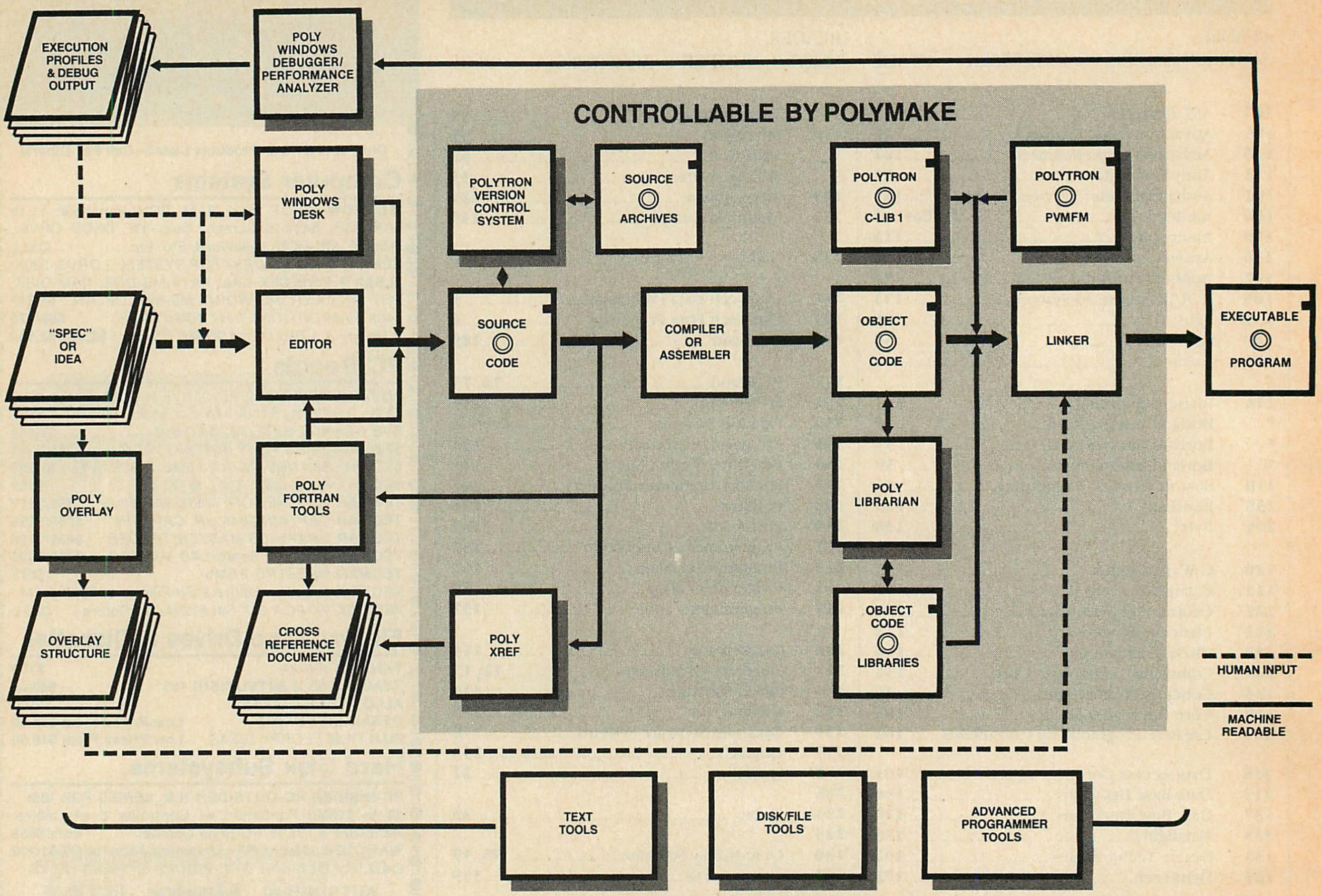
Visa, Mastercard, Checks or C.O.D.
CA residents add 6.5% sales tax
No shipping charges

STERLING CASTLE
702 Washington Street Suite 174
Marina Del Rey, California 90291
(213)306-3020

CIRCLE NO. 183 ON READER SERVICE CARD

INDEX TO ADVERTISERS

POLYTRON TOOLS IN THE SOFTWARE DEVELOPMENT PROCESS



POLYTRON TOOL KITS

Thousands of Professional Software Developers with demanding deadlines turn to high performance MS-DOS/PC-DOS tools from POLYTRON to boost their productivity. All the tools shown above are explained in detail in the **POLYTRON Programmer's Catalog.**

Network & Site Licenses Available.

PolyWindows Desk
The Modular, Expandable Desktop Organizer
100% memory-resident desktop organizer for professionals. Configurable so you can have one or more of the following functions available instantly: Hex/Decimal/Binary Calculator, Standard Calculator, ASCII Table, Multiple Document Text Editor, PolyKey for creating your own keyboard macros, Rolodex-type files, Calendar, Alarm Clock, Appointment Books, Autodialer, Grabber for cut-and-paste between applications. **Copy-Protected $49.95, Not Copy-Protected $84.95.**

PolyWindows Debugger/ Performance Analyzer
A Professional, Memory-Resident Software Development Tool
Instantly analyze and debug ANY program running on your PC anytime. Superior to the IBM Professional Debugger and easier to use. Supports multiple breakpoints, multiple display formats, memory searches, single-step execution, disassembly, hex arithmetic, and block moves. Tells you where your program is spending time and allows you to optimize performance. You can specify the memory ranges for analysis. Results are displayed graphically or in tables. **Includes PolyWindows Desk (Not Copy-Protected) all for only $149.** (Present PolyWindows users can receive trade-in discounts.)

POLYTRON Version Control System (PVCS)
Efficiently maintains revision history of source files for software project management. Maintains chronological, historical records of changes as "edit scripts" (reverse deltas) with full text of the latest "checked in" revision. Reconstructs any prior "revision" of any module, defines a "version" as specified revisions of various modules, supports branching from prior revisions, optional password protection. Friendly, unobtrusive user interface is based on a unique "syntax guide" concept with context-sensitive help. Designed for single or multiple programmer projects. A powerful tool with mainframe power. **Single-user license: $395.**

PolyXREF
The Multi-Lingual Cross Reference System
Generates Cross Reference listings across single or multiple source files of one or more languages. Tells you exactly where each variable and procedure is defined and everywhere it is referenced. **Package includes PolyXREF engine plus one source language module plus .CRF module: $129, additional source language modules [C, Pascal, Assembler or English]: $49 each. Complete package including all four language modules: $219.**

PolyLibrarian
The Object Module Library Manager
The premier library manager allows you to create, examine and manage libraries. Modules can be listed, added, replaced, extracted, and deleted. You can search for public or external names and even change them. Three operating modes can be intermixed for maximum flexibility. **PolyLibrarian I for Microsoft format libraries: $99, PolyLibrarian II for Microsoft and Intel format libraries: $149.**

PolyMake
The Intelligent Program Builder and Maintenance Tool
Automates the software maintenance process. Determines which files (programs, etc.) are out of date and invokes your compiler, linker, librarian and does whatever is necessary to bring your entire system up to date. Frees you from the need to remember which files depend on others and which files have been modified. Remembers the exact sequence of operations necessary to create a new revision. Completely automatic unlike competing products. Once you use it, you can't live without it. **Only $99.**

POLYTRON C Library 1
Over 65 High Performance Routines For Lattice C Compiler Users
These are routines for serious programmers that need Executive and I/O functions. Complete source code (mostly Assembler) included. **Only $99.**

POLYTRON Virtual Memory File Manager (PVMFM)
Provides efficient virtual file access and buffering for both text and binary files with automatic buffer-to-disk swapping. Reduces program access time and handles data structures larger than memory allows. PVMFM is a library of user-callable functions that may be linked with your applications code to provide virtual file management capability and random access file buffering. Compatible with Assembler and high-level languages. **Only $199.**

To Order Products Call
1-800-547-4000
Ask for Dept. No. 310
Foreign & Oregon orders call
(503) 684-3000
Send Checks, P.O.s To:
POLYTRON Corporation
P.O. Box 787 DS-310
Hillsboro, OR 97123
Add $5.00 Shipping To Total Order

To Order detailed Programmer's Catalog, send requests to address above for fast response (or circle reader service card number).

POLYTRON

High Quality Software Since 1982

CIRCLE NO. 159 ON READER SERVICE CARD

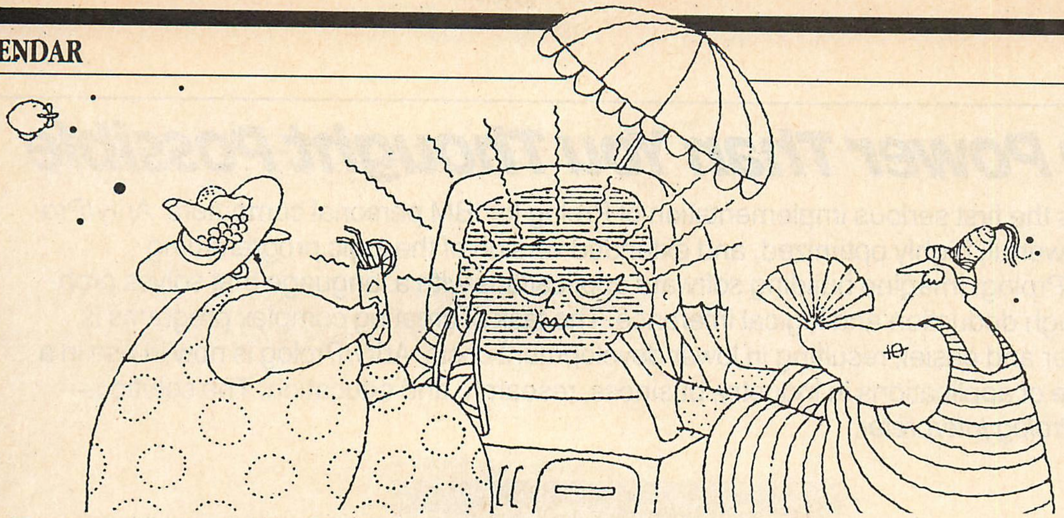


ILLUSTRATION • DAVID POVILAITIS

AUGUST

August 1-3
The Electronic Library/ Automating Education
Dallas, TX
Contact: Infomart, 1950 Stemmons Freeway, Dallas, TX 75207

August 12-14
Data Communications: Network Design, Integration, and Applications
Los Angeles, CA
Contact: Software Institute of America, 8 Windsor Street, Andover, MA 01810; 617/470-3880

August 20-23
14th International Conference on Parallel Processing
St. Charles, IL
Sponsor: The Pennsylvania State University, IEEE, ACM
Contact: IEEE-CS, P. O. Box 639, Silver Spring, MD 20901

August 28-30
8th International Conference on Software Engineering
London, England
Sponsor: British Computer Society, IEE, IEEE-CS with ACM SIGSOFT
Contact: 8th ICSE, IEEE-CS, P. O. Box 639, Silver Spring, MD 20901

SEPTEMBER

September 1
Call for papers: 10th World Computer Conference
Dublin, Ireland
(September 1-5, 1986)
Sponsor: International Federation for Information Processing
Contact: Philip H. Dorn, Dorn Computer Consultants, Inc., 25 E. 86th Street, New York, NY 10028; 212/427-7460

September 3-6
OASI Third Annual Conference: The Integrated Office—How Soon?
Bloomington, MN
Sponsor: Office Automation Society
Contact: OASI, 2108 C Gallows Road, Vienna, VA 22180; 703/790-0490

September 5-7
3rd Personal Computer Faire
San Francisco, CA
Contact: David Small, Computer Faire, Inc., 181 Wells Avenue, Newton, MA 02159; 617/965-8350

September 9-11
Federal Computer Conference
Washington, DC
Contact: The Federal Computer Conference, P. O. Box N, Wayland, MA 01778; 617/358-5356 or 804/747-6448

September 10-13
9th Data Communications Symposium
Whistler Mountain, British Columbia, Canada
Sponsor: ACM SIGCOMM, IEEE-CS, IEEE Communications Society
Contact: W. P. Lidinsky, Room 6B 309, AT&T Bell Laboratories, Naperville Wheaton Road, Naperville, IL 60566; 312/979-6817

September 11-13
CAD 2001: The Countdown
London, England
Contact: CAD Seminars, Inc., 150 E. Riverside, Suite 400, Austin, TX 78704; 512/445-7342

September 16-18
C Language Seminar
Cambridge, MA
Contact: Beatrice Blatteis, CL Publications, 131 Townsend Street, San Francisco, CA 94107; 415/957-9353

September 17-19
Sixth Annual SOFTWARE/expo
Dallas, TX
Contact: SOFTWARE/expo, Suite 205, 2400 E. Devon Avenue, Des Plaines, IL 60018; 312/299-3131

September 18-20
UNIX EXPO
New York, NY
Contact: Robert P. Birkfeld or Don Berey, National Expositions Co., Inc., 14 W. 40th Street, New York, NY 10018; 212/391-9111

September 21-22
7th Annual FORTH National Convention
Palo Alto, CA
Contact: FORTH Interest Group, P. O. Box 8231, San Jose, CA 95155; 408/277-0668

September 24-26
AI/Europa: Artificial Intelligence and Fifth Generation Computer Technology Conference and Exhibition
Wiesbaden, West Germany
Contact: Jim Hay, Tower Conference Management Co., 331 W. Wesley Street, Wheaton, IL 60187; 312/668-8100

September 26-29
8th Northeast Computer Faire
Boston, MA
Contact: David Small, Computer Faire, Inc., 181 Wells Avenue, Newton, MA 02159; 617/965-8350

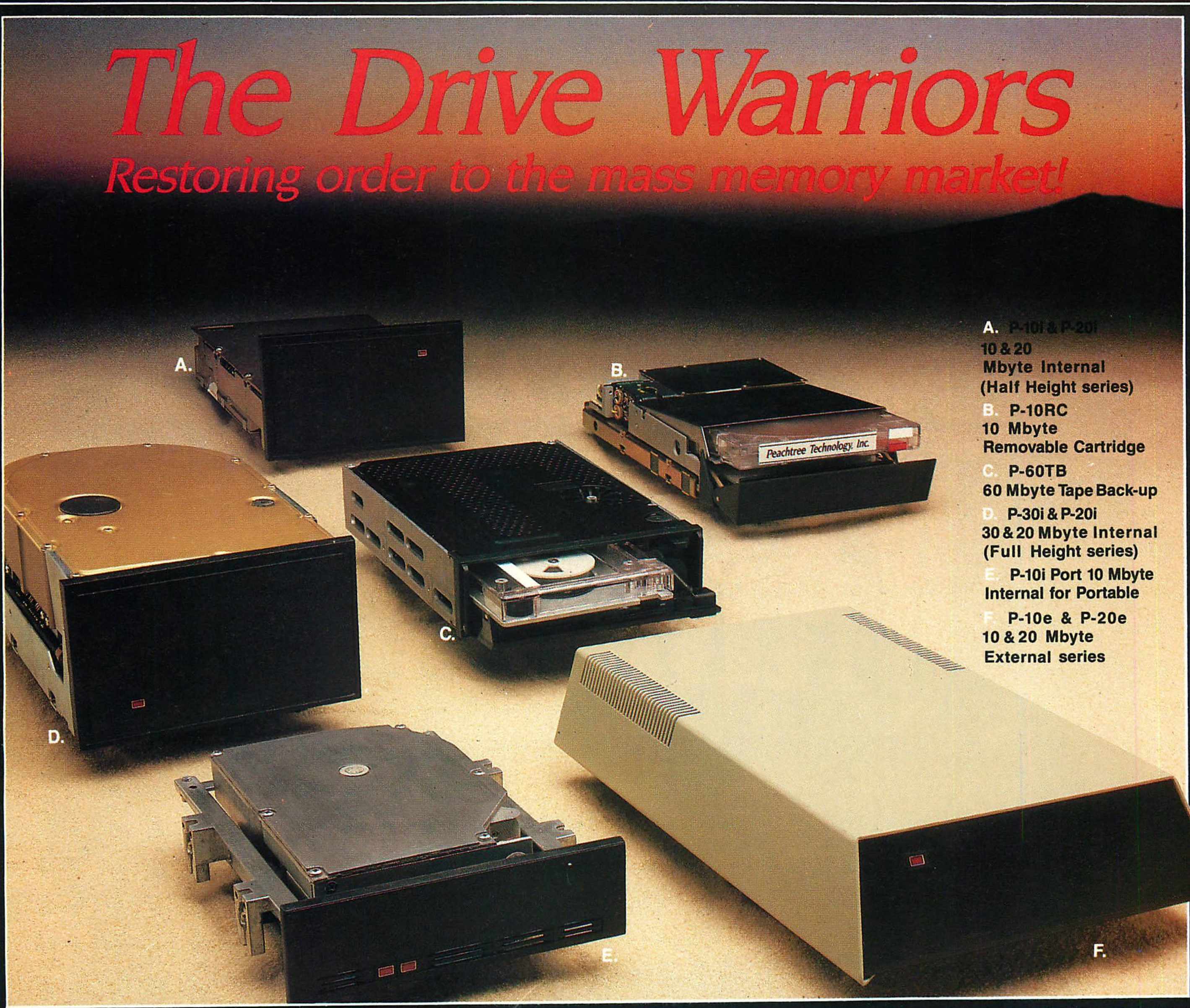
OCTOBER

October 2-4
First International Conference on Optical Storage Technology
Tokyo, Japan
Sponsor: TOC and USACO Corporation
Contact: Technology Opportunity Conference, P.O. Box 14817, San Francisco, CA 94114; 415/626-1133

October 3-5
UNIX Systems Expo 85
Boston, MA
Contact: David Small, Computer Faire, Inc., 181 Wells Avenue, Newton, MA 02159; 617/965-8350

October 5-8
5th Symposium on Small Computers in the Arts
Philadelphia, PA
Contact: Symposium, P. O. Box 1954, Philadelphia, PA 19105

October 14-16
CAD/CAM
Dallas, TX
Contact: Informart, 1950 Stemmons Freeway, Dallas, TX 75207



The mass memory market has become a real battle zone! Chaos, confusion and buyer uncertainty have resulted from too many products and not enough answers.

The truth of the matter is, anyone can sell a winchester disk drive. The problem is that hard drives, by their own nature, are sensitive devices that require appropriate handling and customer support.

Don't be mislead into believing that a disk drive can be sold and booted around the country, without any risk.

There is no room for the weak, when it comes to delivering a product you can depend on!

Peachtree Technology Inc. has cleared the path of confusion with a comprehensive line of mass memory expansion systems that have established themselves as leaders in advanced hard drive technology, backed by strong customer support. Every Peachtree Technology Inc. product has been through stringent diagnostic testing as a complete system. To assure they not only meet, but surpass specifications. (On the outside of every box is a documented certificate of inspection signed by our technician). Plus we stand behind our products with a complete one year parts & labor warranty.

FLEXIBILITY AND COMPATIBILITY A MUST! To eliminate frustration, Peachtree Technology Inc. has provided a complete line of mass memory storage systems. They are fully compatible with IBM PC, XT, & PORTABLE, AT&T, CORONA PORTABLE, COMPAQ DESK PRO & PORTABLE, COLUMBIA, TAVA, ZENITH 150 AND MANY MORE. (If your computer is not listed, please contact us for further information).

We offer a complete line of products including:

- External winchester hard drives • 10, 20, and 30 megabyte (formatted capacity). Our 10 and 30 megabyte drives utilize a hard carbon coating on the disk platter surface.
- Internal mounted winchester hard drives • 10, 20 and 30 megabyte (formatted capacity)
- Removable hard disk • Half height winchester, 10 megabyte, utilizing new and advanced head positioning technology.
- Cartridge tapesystems • Capacity – up to 60 mbytes • Half high 5.25" form factor add-in/add-on to IBM-PC and compatibles.
- A 135 watt internal power supply that replaces the standard unit in the PC.

We'll supply everything you need to give your micro new speed and storage capacity: Disk drive, controller, cables, software, and associated custom firmware.

One stop shopping for your mass memory requirements.

Don't become a victim! The frustration and inconvenience of not having customer support may not be worth the few dollars you save buying from an unreliable source. There is no substitute for a complete hands on demonstration and face to face commitment to customer service from a reputable dealer. So don't be left standing in the smoke. To find out more about Peachtree Technology Inc. and our complete line of memory expansion systems, ask your local computer dealer for a complete demonstration. If he does not have our products have him call 404-662-5556, and we will deliver a demonstration unit & literature within 24 hours!

CIRCLE NO. 158 ON READER SERVICE CARD

Peachtree Technology, Inc.

3020 Business Park Drive, Norcross, Georgia 30071, Telephone (404) 662-5158 Sales (404) 662-5556

IBM – A Registered Trademark of International Business Machines Inc. AT&T – A Registered Trademark of Bell Laboratories. Corona – A Registered Trademark of Corona Data Systems. Compaq – A Registered Trademark of Compaq Computer Corp. Columbia – A Registered Trademark of Columbia Data Products. Tava – A Registered Trademark of Tava Corp. Zenith – A Registered Trademark of Zenith Data


2-YEAR
WARRANTY
AMDEK
COLOR 700
BRT
CNT
TXT
POWER